OBJECT-ORIENTED TEXTURE ANALYSIS AND UNSUPERVISED SEGMENTATION FOR HISTOPATHOLOGICAL IMAGES

A DISSERTATION SUBMITTED TO
THE DEPARTMENT OF COMPUTER ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Akif Burak Tosun
August, 2012

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Assist. Prof. Dr. Çiğdem Gündüz Demir(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Prof. Dr. H. Altay Güvenir

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Prof. Dr. Enis A. Çetin

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Prof. Dr. Cevdet Aykanat

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Assist. Prof. Dr. Aybar Acar

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

ABSTRACT

OBJECT-ORIENTED TEXTURE ANALYSIS AND UNSUPERVISED SEGMENTATION FOR HISTOPATHOLOGICAL IMAGES

Akif Burak Tosun
Ph.D. in Computer Engineering
Supervisor: Assist. Prof. Dr. Çiğdem Gündüz Demir
August, 2012

The histopathological examination of tissue specimens is essential for cancer diagnosis and grading. However, this examination is subject to a considerable amount of observer variability as it mainly relies on visual interpretation of pathologists. To alleviate this problem, it is very important to develop computational quantitative tools, for which image segmentation constitutes the core step. The segmentation algorithms in literature commonly use pixel-level color/texture descriptors that they define on image pixels for quantizing a tissue. On the other hand, it is usually harder to express domain specific knowledge about tissues, such as the spatial organization of tissue components, using only the pixel-level descriptors. This may become even harder for tissue images, which typically consist of a considerable amount of variation and noise at their pixel-level, such as similar color distribution of different tissue components, distortion in cell alignments, and color contrast caused by too much stain in a particular region. The previous segmentation algorithms are more susceptible to these problems as they work on pixel-level descriptors.

In order to successfully address these issues, in this thesis, we introduce three new texture descriptors, namely ObjSEG, GraphRLM, and ObjCooc textures, and implement algorithms that use these descriptors for segmenting histopathological tissue images. We extract these texture descriptors on tissue components that are approximately represented by circular objects. Since these object-oriented texture descriptors are defined on the tissue components, and hence domain specific knowledge, they represent the spatial organization of the components better than their previous counterparts. Thus, our algorithms based on

these descriptors give more effective and robust segmentation results. Furthermore, since the descriptors are not directly defined on image pixels, they are effective to alleviate the pixel-level problems.

In our experiments, we tested our algorithms that use the proposed object-oriented descriptors on a dataset of 200 colon tissue images. Our experiments demonstrated that our new object-oriented feature descriptors led to high segmentation accuracies, also providing a reasonable number of segmented regions. Compared with its previous counterparts, the experimental results also showed that our proposed algorithms are more effective in segmenting histopathological images.

Keywords: Image segmentation, Texture analysis, Histopathological image analysis, Graphs, Object-oriented texture, Cancer.

ÖZET

HİSTOPATOLOJİK İMGELERDE NESNEYE DAYALI DOKU ANALİZİ VE ÖĞRETİCİSİZ BÖLÜTLEME

Akif Burak Tosun
Bilgisayar Mühendisliği, Doktora
Tez Yöneticisi: Assist. Prof. Dr. Çiğdem Gündüz Demir
Ağustos, 2012

Kanser tanısı ve derecelendirilmesi için doku örneklerinin pataloglar tarafından incelenmesi gereklidir. Fakat, bu inceleme patalogların dokuyu görsel olarak incelemesinden dolayı oldukça fazla öznelliğe sebep olur. Bu problemi azaltmak için bilgisayar ortamında ölçülebilir değerler ile çalışan sistemler geliştirilmelidir ve imge bölütleme işlemi de bu sistemlerin temel taşıdır. Literatürdeki imge bölütleme yöntemleri dokuları nicelendirmek için genel olarak piksel düzeyindeki renk/yapı tanımlayıcılarını kullanır. Ancak piksel düzeyindeki bu bilgiler doku bileşenlerinin uzamsal organizasyonu gibi patolojiye özgü bilgiyi ifade edebilmek ve bu bilgiyi kullanabilmek için yeterli değildir. Bunun yanında, genel olarak doku resimlerinde oldukça fazla varyasyon ve piksel düzeyinde parazit bulunur; farklı doku bileşenlerinin benzer renk dağılımları, hücre dizilimlerindeki dağılmalar ve fazla boyanmadan dolayı oluşan bölgesel renk karşıtlıkları gibi. Önceki bölütleme yöntemleri piksel düzeyinde tanımlayıcılar kullandıkları için belirtilen bu problemlere karşı daha fazla duyarlıdır.

Bu sorunları başarılı bir biçimde çözebilmek için, bu tezde üç yeni yapısal tanımlayıcı sunduk – bunlar ObjSEG, GraphRLM ve ObjCooc yapıları olarak adlandırıldı – ve histopatolojik doku resimlerinin bölütlenmesi için bu tanımlayıcıları kullanan yöntemler geliştirdik. Bu tanımlayıcıları yaklaşık olarak dairesel nesnelerle betimlenmiş doku bileşenleri üzerinde hesapladık. Belirtilen nesneye dayalı tanımlayıcılar doku bileşenleri üzerinden çıkarıldıkları için bu bileşenlerin uzamsal organizasyonunu ve dolaylı olarak patolojiye özgü bilgileri literatürdeki emsallerine kıyasla daha iyi temsil edebilirler. Böylece bu tanımlayıcıları kullanarak geliştirdiğimiz yöntemler de daha verimli ve güçlü sonuçlar çıkarabilır. Ayrıca, bu tanımlayıcılar direk olarak imge pikselleri üzerinden hesaplanmadıkları için piksel düzeyindeki problemleri de azaltma konusunda daha etkilidirler.

Deneylerimizde, sunduğumuz nesneye dayalı tanımlayıcıları kullanan imge bölütleme yöntemlerimizi 200 kolon doku imgesinde test ettik. Deneylerimiz, nesneye dayalı bu yeni tanımlayıcıların hem yüksek oranda doğruluk veren bölütleme sonuçları çıkardığını hem de bölütlenmiş alan sayısını makul seviyelerde tutmayı başardığını kanıtladı. Deneysel sonuçlarımız, önceki imge bölütleme yöntemleriyle de karşılaştırıldığında, sunduğumuz yöntemlerin histopatolojik resimlerin bölütlenmesinde daha etkili olduğunu gösterdi.

Anahtar sözcükler: İmge bölütleme, Doku analizi, Histopatolojik imge analizi, Çizgeler, Nesneye dayalı doku, Kanser.

Acknowledgement

I would like to give my greatest thanks to my advisor Assist. Prof. Dr. Çiğdem Gündüz Demir for her endless support, instructive comments, suggestions, and encouragement during this thesis study. Without her guidance and patience I would probably lost my way in this research and this thesis would have never been completed.

I would like to thank my thesis committee members Prof. Dr. H. Altay Güvenir, Prof. Dr. Enis A. Çetin, Prof. Dr. Cevdet Aykanat and Assist. Prof. Dr. Aybar Acar for their valuable comments and suggestions to improve this thesis.

I also would like to thank Prof. Dr. Cenk Sökmensüer for his guidance. Without his support I will not be able to understand the medical terminology and the importance of my thesis study.

My graduate study had been financially supported by the Computer Engineering Department of Bilkent University and The Scientific and Technological Research Council of Turkey (TÜBİTAK). I would like to thank both institutions for their support.

I also thank my collegues in our research group for their cooperation during this study and for their friendship.

I am deeply thankful to my parents Semra and Dr. Mehmet Tosun, my sister Neslihan, who supported me in each and every day.

Finally, I would like to thank my wife Ayça for her endless support and patience (and for the snacks of course) while I spent hours in front of my computer during this thesis study.

To my family, my lovely wife Ayça and my baby girl Defne...

Contents

1 Introduction **1**

1.1 Motivation 2

1.2 Contribution 4

2 Background **11**

2.1 Medical background 11

2.2 Related works on image segmentation 14

3 Object definition **21**

3.1 Clustering image pixels 21

3.2 Preprocessing 23

3.3 Circle-fitting 25

4 ObjSEG **29**

4.1 Object-oriented texture definition 30

4.2 Segmentation algorithm 32

4.2.1 Seed-determination 34
4.2.2 Seed-growing 34
4.2.3 Region-merge 36
4.3 Experiments 37
4.4 Modified region growing algorithm for ObjSEG 45
4.4.1 Object-to-object relationship definition 45
4.4.2 Region growing algorithm 48
4.4.3 Experiments 49
4.4.4 Summary 53

5 GraphRLM **54**

5.1 Graph generation 54
5.2 Run-length matrix calculation 56
5.3 Feature extraction 57
5.4 Segmentation algorithm 59
5.5 Experiments 61
5.5.1 Evaluation 62
5.5.2 Comparisons 62
5.5.3 Parameter selection 64
5.5.4 Test results 72
5.5.5 Parameter analysis 73

5.5.6 Robustness analysis 77

5.5.7 Computational time analysis 78

5.6 Summary 78

6 Segmentation by object co-occurrence 80

6.1 Feature extraction 80

6.2 Segmentation 84

6.3 Experiments 85

6.3.1 Evaluation 86

6.3.2 Results 86

6.3.3 Comparisons 87

6.3.4 Discussion 89

6.4 Summary 95

7 Conclusion 97

List of Figures

1.1 Segmentation of homogeneous regions from a low-magnification heterogeneous image. 2

1.2 Illustration of the methodology followed by our studies. 6

1.3 Sliding window examples for ObjSEG homogeneity computation. 7

1.4 Illustration of computing (a) gray-level run-length matrices and (b) graph run-length matrices. 8

1.5 Illustration of computing (a) gray-level co-occurrence matrices and (b) object graph co-occurrence matrices. 9

2.1 Main cytological components in a colon tissue: cellular, stromal, and luminal components. 12

2.2 Example images of colon tissues stained with hematoxylin-and-eosin, which is routinely used to stain biopsies in hospitals. . . . 13

2.3 Examples of noise, artifacts, and variations caused by misconduct and variation in tissue preparation procedures. 18

3.1 Results of the k-means algorithm on an example histopathological tissue image: (a) original image, (b) clustered pixels, (c) zoomed sub-image, and (d) clustered pixels of the zoomed sub-image. In the cluster maps, white-like, pink-like, and purple-like pixels are shown in cyan, pink, and purple, respectively. 22

3.2 The separation of nuclei (purple-like), lumen (white-like), and stroma (pink-like) pixels in the preprocessing step. 23

3.3 The effect of dilation operation on nuclei, lumen and stroma pixels in preprocessing step. 24

3.4 The results of the circle-fitting procedure on an example sub-image. The objects defined on nuclei, lumen, and stroma pixels are shown in (a), (b), and (c), respectively. All these objects are shown together in (d). 26

3.5 The results of the circle-fitting procedure on a larger image. . . . 28

4.1 An example of a circular window W located at the centroid (X_w, Y_w). 32

4.2 A schematic of the proposed region-growing method. 33

4.3 The steps for the object-oriented segmentation of a biopsy image using ObjSEG 35

4.4 The segmentation results. (a1)-(d1) The manual segmentations; (a2)-(d2) the results obtained by our object-based algorithm; (a3)-(d3) the results obtained by the pixel-based JSEG algorithm. . . 38

4.5 The segmentation results. (e1)-(h1) The manual segmentations; (e2)-(h2) the results obtained by our object-based algorithm; (e3)-(h3) the results obtained by the pixel-based JSEG algorithm. . . 40

4.6 The segmentation results. (i1)-(l1) The manual segmentations; (i2)-(l2) the results obtained by our object-based algorithm; (i3)-(l3) the results obtained by the pixel-based JSEG algorithm. . . 42

4.7 The segmentation results. (m1)-(p1) The manual segmentations; (m2)-(p2) the results obtained by our object-based algorithm; (m3)-(p3) the results obtained by the pixel-based JSEG algorithm. 44

4.8 The illustration of the proposed region growing algorithm for relational objects 46

4.9 An example of a circular window W located at object centroid (X_i, Y_i). 47

4.10 The visual results of the object-based growing segmentation algorithm (a1)–(d1) and the pixel-based growing (ObjSEG) algorithm (a2)–(d2). Segmented regions are shown with different colors. The manual segmentations are also indicated in these images. 51

4.11 The visual results of the object-based growing segmentation algorithm (e1)–(h1) and the pixel-based growing (ObjSEG) algorithm (e2)–(h2). Segmented regions are shown with different colors. The manual segmentations are also indicated in these images. 52

5.1 A graph generated for representing the spatial distribution of cytological components within a tissu 55

5.2 The graph of the subimage confined within a rectangle in Fig. 5.1. 56

5.3 Illustration of calculating a graph run-length matrix 57

5.4 The illustration of the segmentation algorithm (GraphRLM) . . . 60

5.5 The segmentation accuracy and the number of segmented regions as a function of the merge threshold (minimum area) parameter. These results are obtained on the training samples for (a) GraphRLM, (b) GrayRLM, (c) JSEG, (d) GBS, and (e) ObjSEG algorithms. 66

5.6 The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered. 68

5.7 The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered. 69

5.8 The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered. 70

5.9 The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered. 71

5.10 The segmentation accuracy and the number of segmented regions as a function of the model parameters: (a) window size, (b) distance threshold, (c) component size threshold, (d) merge threshold, and (e) grow rate percentage. 75

5.11 The effects of the contrast change ratio on (a) the segmentation accuracy and (b) the number of segmented regions. 77

6.1 An illustration of object co-occurrence matrix definition for a selected white object shown as a thick-bordered cyan circle. The window $\mathcal{W}$ is located at the center object but the steps illustrated here are repeated for every object in $\mathcal{W}$. Here we select a side object for illustration since we want to show the definition for larger distances (for this side object, the maximum distance is 7, which would be smaller if for example the center object was selected). . 82

6.2 Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3. 90

6.3 Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3. 91

6.4 Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3. 92

6.5 Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3. 93

6.6 The visual results obtained on the example images of tissues stained with immunohistochemistry. 95

List of Tables

4.1 The average and the standard deviation of the sensitivity, specificity, and accuracy percentages for the proposed object-based and the pixel-based algorithms. 44

4.2 The average and the standard deviation of the sensitivity, specificity, and accuracy percentages for the object-based and the pixel-based growing algorithms. 50

5.1 Texture features for gray-level run-length matrices. 58

5.2 The parameters of the algorithms and their values that are considered in the estimation of the best parameter sets. 63

5.3 The average and standard deviation of segmentation results obtained on the training samples. Parameter sets are selected on the training samples without any restriction on the number of the segmented regions. 65

5.4 The average and standard deviation of segmentation results obtained on the training samples. Parameter sets are selected on the training samples considering only the parameter combinations that give at most (a) 5 regions and (b) 10 regions. 67

5.5 The average and standard deviation of segmentation results obtained on the test samples. Parameter sets are selected on the training samples considering only the parameter combinations that give at most 10 regions. 73

5.6 Dissimilarity measures analyzed in the experiments. 76

5.7 The effects of a dissimilarity measure on the segmentation accuracy and the number of segmented regions. 76

5.8 Computational times of the algorithms. 78

6.1 The average test set results obtained by the proposed *ObjCooc* algorithm and their standard deviations. 87

6.2 The average test set results obtained by the other algorithms and their standard deviations. The parameters of the algorithms are selected considering an upperbound N on the number of segmented regions. 88

6.3 The average test set results obtained by the *ObjCooc*, *GraphRLM*, *ObjSEG*, and *GBS* algorithms and their standard deviations. . . . 89

6.4 The average test set results obtained by the *ObjCooc* algorithm that uses an alternative co-occurrence matrix calculation. 94

Chapter 1

Introduction

In the current practice of medicine, there has been an increasing use of imaging systems in making decisions for several medical phenomena. Since most of the decisions are made by visual examination of the patient, medical imaging systems attract most of the attention. There are many types of imaging techniques for different parts of human body; such as magnetic resonance imaging for brain analysis, ultrasound imaging for fetal echocardio analysis, and microscopic images for tissue analysis. As a result, there is a huge amount of research done in medical imaging and many researchers in computer science focus on medical imaging technologies. However, there is not much effort on medical image analysis and usually the images are not processed for decision making before serving them to medical experts. Instead, the experts make their decisions by visually examining the raw images [1].

The expert examination of medical images is very time consuming in many fields of medicine, such as histopathological examination of cancerous tissues. It also decreases the performance of the expert since they have to examine a huge amount of images in a single day. This performance drop affects their decisions and increases the possibility of making a wrong decision. To avoid this problem, computer-aided medical image analysis techniques should be applied to raw medical images and experts should work on preprocessed and selected images. It is expected that, with the help of intelligent medical image analysis systems

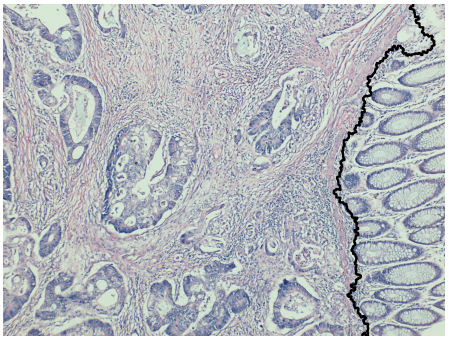
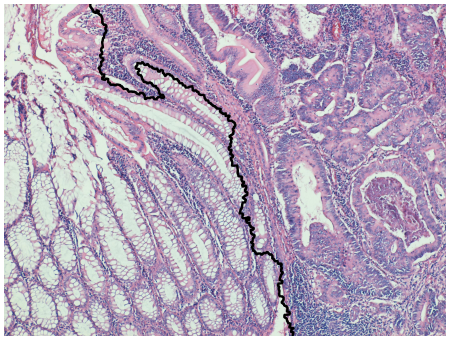

(a)

(b)

Figure 1.1: Segmentation of homogeneous regions from a low-magnification heterogeneous image.

medical experts can make their decisions more accurately.

1.1 Motivation

Medical image analysis can be used for different applications such as locating tumors and other pathologies, measuring tissue volumes, diagnosis, treatment planning, computer-guided surgery, and studying the anatomical structure of human body. In this thesis, we focus on medical image analysis in the field of *histopathology*. Histopathology is the study of diseased tissues in clinical medicine. It is performed by a pathologist on sections of a tissue removed from a diseased or abnormal part of the body. The main purpose of this province is to locate and grade cancerous (malignant) regions on the tissue by visually examining tissue sections under a microscope. After locating cancerous regions, the pathologist determines the cancer type and its grade according to the changes in cell morphology and cellular distribution in tissues.

In our case, pathologists are searching for a better and effective tool that helps their decision making. They think of an automated system that locates cancerous regions on a tissue slide and then makes primary examination on these regions. Finally, they will examine only the selected regions and make the final decision. Such a system would save a lot of time, which leads to faster and effective diagnosis. Moreover, since these systems would work faster, more people can be

routinely controlled and early diagnosis can be made to save more lives.

Automated analysis of histopathological tissue images not only increases throughput but also improves reproducibility. Digital pathology systems have been implemented for different purposes including classification [2, 3, 4], retrieval [5, 6], and segmentation, which can further be categorized into two in terms of its objective. Cell/gland segmentation [7, 8] aim to locate cells and glands on a tissue image whereas tissue image segmentation, which is the focus of this thesis, aims to divide a heterogeneous tissue image into its homogeneous regions. There exist only few studies on tissue image segmentation and most of them use grid analysis for segmentation [9, 10, 11, 12]. There is a need for a tissue image segmentation methodology that uses medical background information in its feature definitions in order to represent the tissues better.

Early diagnosis and correct grading of a cancerous specimen is very vital for the selection of a treatment plan. However, especially grading is very subjective due to the human factor [13, 14, 15, 16]. This subjectivity may become perceptible in different decisions of different pathologists on the same tissue (inter-observer variability) as well as in different decisions of the same pathologist on the same tissue at different times (intra-observer variability). The subjectivity problem is vital since misdiagnosis of a patient/disease leads to delayed treatment whereas misgrading may result in selecting an incorrect treatment plan. On the other hand, with early diagnosis and correct treatment of cancer, high survival rates could be achieved (e.g., for colorectal cancer, with early diagnosis and correct treatment, survival rates reach up to 97 percent [17]). To overcome the subjectivity problem, it is very important to develop objective methods that rely on mathematical features (measures).

Currently, researchers in medicine have been working on reducing this subjectivity by adding new criteria and refining the grading schemes frequently [18, 19]. Recently defined criteria are usually based on quantitative measurements extracted from the specimen such as the percentage of the cells belonging to a tissue structure and the percentage of cells in a particular area [20]. However, these grading schemes are not practically applicable by the pathologists. The

main reason behind this is the fact that the quantitative measures are calculated with the visual assessment of the experts. For instance, in cancer grading, each grade is characterized by the degree of distortions and irregularities observed in a biopsy tissue [21]. Although these degrees are defined quantitatively, the quantification should be done by the eyes of a pathologist.

Towards this end, there is a huge amount of efforts that propose to use different computational approaches for extracting mathematical features from tissue images [3, 22, 23, 24, 25, 26, 27] and some of these approaches especially focus on defining mathematical features from colon images [28, 29, 30, 31, 32]. These computational methods extract a set of mathematical features (e.g., morphological [33, 34], textural [22, 23, 35, 36, 37], fractal [30, 38], and structural [24, 25, 36, 39, 40] features). Most of these studies make their analyses on tissue images by extracting three features from tissue images assuming that they are homogeneous. On the other hand, images are usually heterogeneous in the real life and these heterogeneous images should be segmented into their homogeneous regions before applying these computational methods. However, obtaining a dataset of homogeneous images is hard and time consuming. Therefore, image segmentation is at the heart of these computational methods.

1.2 Contribution

Our aim is to segment heterogeneous histopathological images for finding homogenous regions as in Fig. 1.1. As a result of image segmentation, the segmented homogeneous regions can medically be interpreted in the same way so that the aforementioned feature extraction methods could be used directly. There are many studies proposed in the literature to segment heterogeneous images [41, 42, 43, 44].

However, most of these studies are general purpose segmentation algorithms and they are not specifically designed for histopathological images. Thus, they do not consider the background knowledge of tissue organization to define their features for segmentation process. Moreover, they usually use low-level (pixel-level)

information in their segmentation steps, which is usually the main reason behind unsuccessful segmentations in histopathological images. During the preparation and staining routines of histopathological specimens in hospitals, there is always a risk of error due to the human factor. This error includes noise, distortions in cell alignments, color contrast caused by too much stain in a particular region, etc. In this thesis, we aim at solving the problems caused by low-level information by introducing new texture definitions for histopathological images and use these definitions in segmentation. The methodology followed for this purpose is illustrated in Fig. 1.2.

A tissue is characterized by the distribution of its cytological components. In this thesis, we propose to use high-level information that is extracted from the distribution of cytological components of a tissue (Fig. 2.1). For this purpose, we decompose an image into its cytological components. In the ideal way, these components should be segmented; however, this segmentation is a very difficult problem (even for a human eye) because of the complex nature of a histopathological image and the magnification in which the segmentation is achieved. Therefore, we approximately represent these components (and hence, the tissue) by defining a set of primitive circular objects. The details of this object definition is given in Chapter 3.

This object definition allows us to define textural criteria using the high-level information of tissue component distribution and use these criteria in the segmentation of histopathological images. In this thesis, we introduce three new textural feature definitions, namely ObjSEG [45], GraphRLM [46], and ObjCooc [47].

In our first approach, ObjSEG [45], we introduce a new homogeneity measure and demonstrate a new segmentation algorithm that uses this homogeneity measure to segment a biopsy image. Our proposed approach relies on the quantification of the spatial relations between the tissue components (e.g., epithelial tissue components, connective tissue components, and luminal structures). For this purpose, we define different types of "objects", which represent different components of a tissue, and make use of the distribution of these objects as well as their spatial relations to define our homogeneity measure. This measure simply

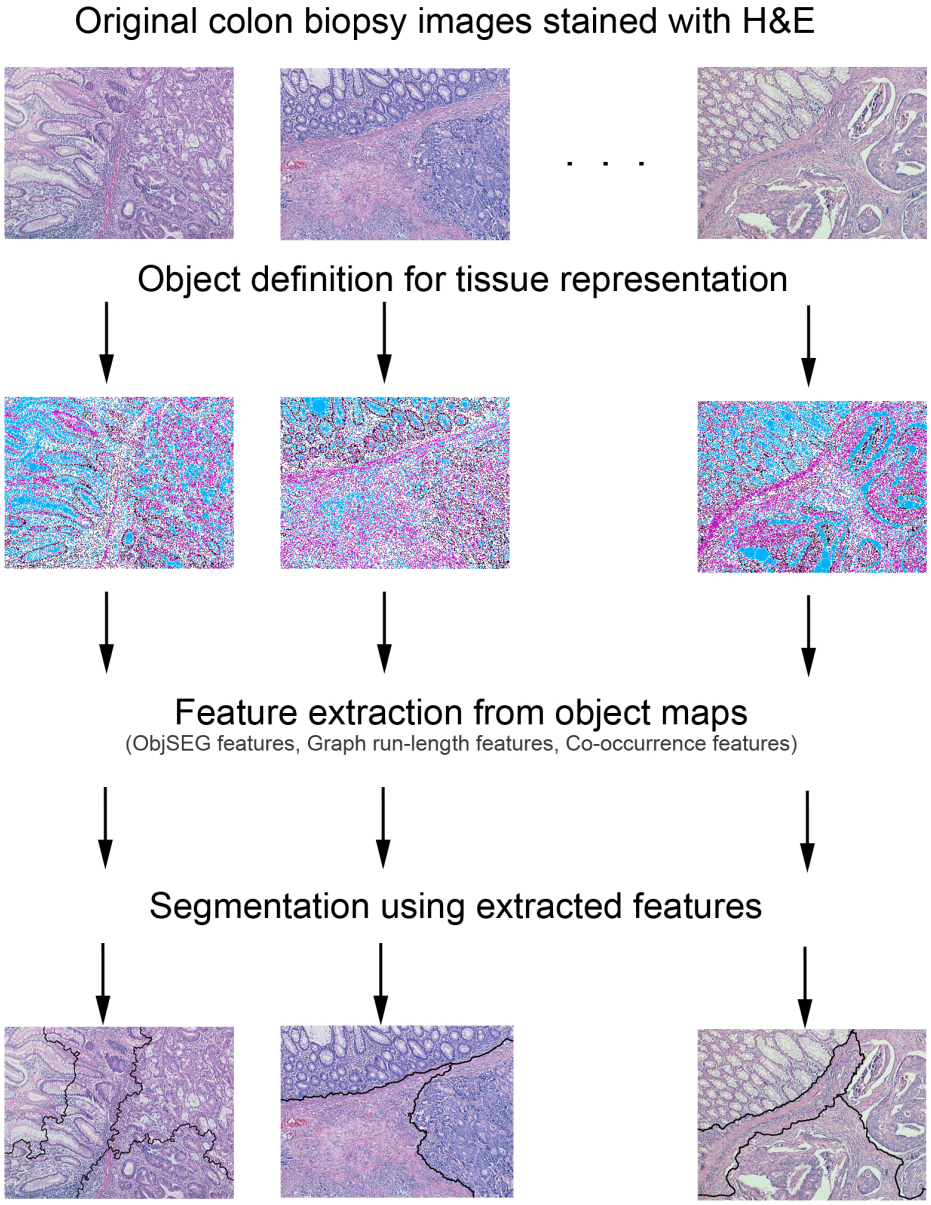


Figure 1.2: Illustration of the methodology followed by our studies.

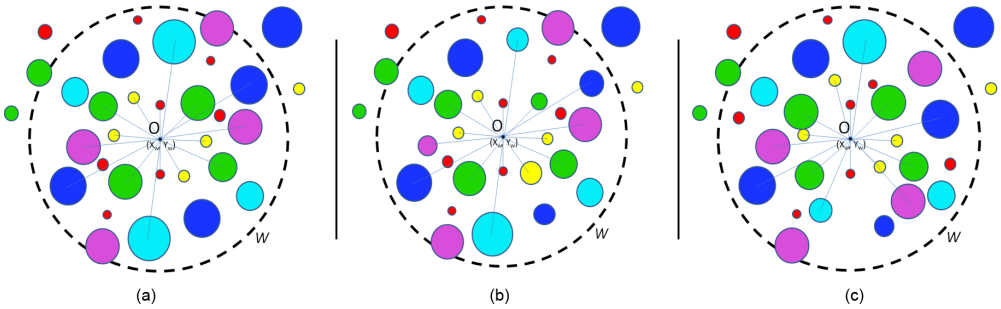
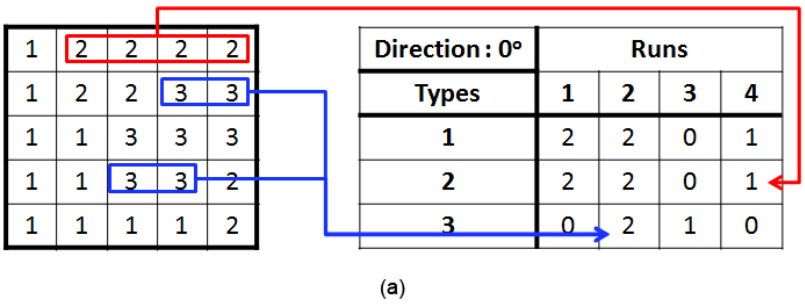


Figure 1.3: Sliding window examples for ObjSEG homogeneity computation.

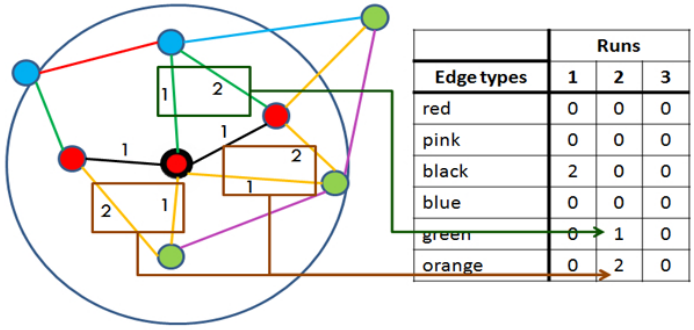
uses the observation that for an object in a circular window W, in which the measure is calculated, there should be another one with the same type and the same size on the opposite direction with reference to the window center $O(X_w, Y_w)$. Some examples are given in Fig. 1.3. The window W shown in Fig. 1.3(a) is homogeneous with respect to this definition since every object has a corresponding object with the same size and the same type on the other side of the window center $O(X_w, Y_w)$. The window in Fig. 1.3(b) is less homogeneous than the window in Fig. 1.3(a), since there are opposing objects of the same type, but some of these objects are smaller than their counterparts. Finally, the window in Fig. 1.3(c) has the least homogeneity value since different types of the objects with different sizes are scattered around the center of the window. The calculation of this measure is detailed in Chapter 4.

In this first approach, the segmentation algorithm uses this object based information, as opposed to the existing algorithms, which rely on using pixel-based information (pixel colors and/or pixel-based textures). This object-oriented algorithm uses region growing algorithm on the features extracted for pixels. Our experiments demonstrated that this algorithm yields higher accuracy on the average and significantly improves the accuracy in locating tumorous regions and other non-cancerous tissue transformations compared to its pixel-based counterparts. The details of this segmentation algorithm are also explained in Chapter 4.

As an extension of this first approach, we propose another region growing algorithm [48], in which the growing process depends on object-to-object relations, instead of pixel connectivity. It is different than the ObjSEG algorithm that



(a)

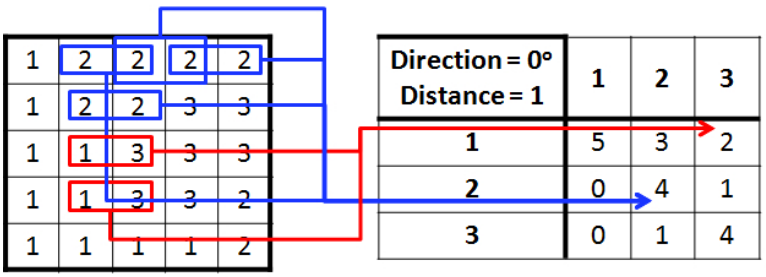


(b)

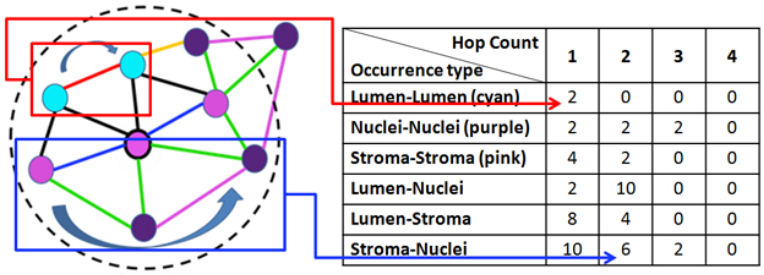
Figure 1.4: Illustration of computing (a) gray-level run-length matrices and (b) graph run-length matrices.

grows regions on pixels. Our experiments show that the use of object-to-object relationships in region growing increases the segmentation performance. It also improves the robustness and speed of the algorithm.

In the second approach, we introduced GraphRLM features that are defined also to model spatial distribution of cytological components [46]. In particular, this approach defines the texture of cytological components on a graph using the idea of gray-level run-length matrices [49]. However, our approach considers the runs of cytological components on the graph to form a run-length matrix, instead of considering the runs of pixel intensities. In other words, the algorithm constructs "a graph run-length matrix" by counting the number of "graph-edge runs" instead of constructing a gray-level run-length matrix by counting the number of gray runs. We illustrate the computation of gray-level run-length matrices and graph run-length matrices in Fig. 1.4(a) and Fig. 1.4(b), respectively. As seen in this example, the graph run-length matrix computation uses edge types instead



(a)



(b)

Figure 1.5: Illustration of computing (a) gray-level co-occurrence matrices and (b) object graph co-occurrence matrices.

of gray values. However, it does not use a direction variable, as in the case of gray-level run-length matrix calculation. In the example shown in Fig 1.4(b), the matrix computation is done only for the object in the center of the circular window. This matrix will be computed for all objects in the window and then accumulated into one matrix in order to represent whole window's run-length matrix. Our experiments demonstrate that a region growing based segmentation algorithm that uses this new texture definition improves the segmentation performance for histopathological images. The details of this algorithm together with its results are given in Chapter 5.

Our final approach, ObjCooc [47], also defines a new set of high-level texture features to represent the background knowledge of spatial organization of tissue components. These texture features are defined on the tissue components, which are approximately represented by tissue objects, and quantify the frequency of two component types being co-occurred in a particular spatial relationship. This

relationship is not defined on pixels of the image as in Fig. 1.5(a), which illustrates the co-occurrence matrix computation on a gray-level image but defined on components, as shown in Fig. 1.5(b). Thus, the object co-occurrence features are expected to be less vulnerable to noise and variations that are typically observed at the pixel-level of tissue images. This feature set is used in a multilevel segmentation algorithm, in which image segmentation is achieved by partitioning the objects according to their co-occurrence features. The details of this algorithm together with its results are given in Chapter 6.

The remainder of this thesis is organized as follows. Medical background is provided and related work in the literature is discussed in Chapter 2. The definition of our object representation is given in Chapter 3. Subsequently, our studies and details about their methodologies are presented in Chapters 4, 5, and 6, together with their experimental results. Finally, a summary of our work is provided together with a discussion about future research perspectives in Chapter 7.

Chapter 2

Background

In this thesis, we focus on the unsupervised histopathological image segmentation of colon tissues. Thus, in this chapter, we first provide medical background information about colon tissues. Then, we will provide a survey of related studies in the context of general unsupervised image segmentation and tissue image segmentation.

2.1 Medical background

Incorporating medical background knowledge, which is specific to the intent of segmentation and the image content, into segmentation will improve the success of image segmentation. In histopathological images, this domain specific knowledge includes the normal appearance of a tissue, which could be expressed in terms of the organization of the cytological tissue components. Cancer causes changes in the organization of these components, leading to tissues deviating from their normal appearances.

In colon tissues, epithelial cells are lined up around a lumen to form glandular structures and non-epithelial cells take place in stroma found in between these glands (Fig. 2.1). Colon adenocarcinomas, which account for 90-95 percent of all

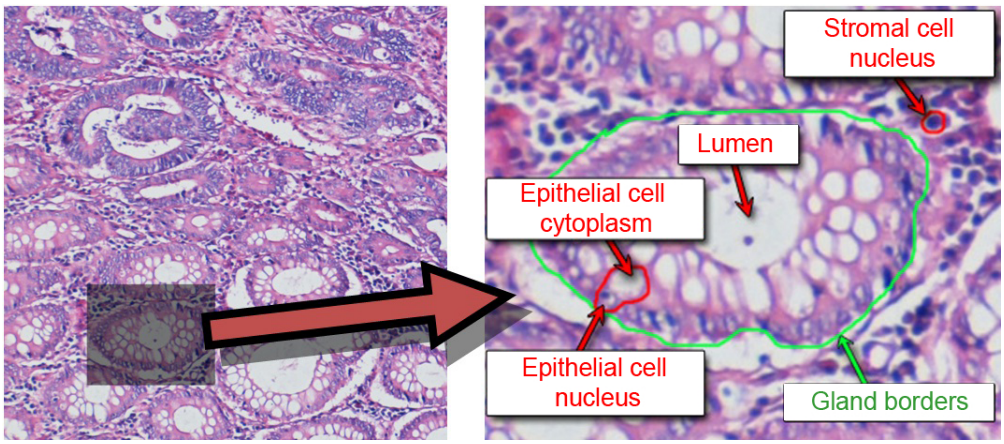


Figure 2.1: Main cytological components in a colon tissue: cellular, stromal, and luminal components.

colorectal cancers that is the second leading cause of cancer-related deaths in the Western world [50], cause organizational changes in colon tissues. Pathologists differentiate normal and cancerous regions by looking at these changes; different kinds of tissue regions are shown in Fig. 2.2.

All tissue images in Fig. 2.2 include both normal regions (marked with 1) and cancerous (adenocarcinomatous) regions (marked with 2). In Fig. 2.2(a) and Fig. 2.2(b), regions marked with 2 are the examples of Grade 1 colon adenocarcinoma. Colon adenocarcinoma originates from the epithelial cells of colon tissues and distorts the normal architecture of colon glands. In Grade 1, the distortion is low so that glands can still be differentiated; here epithelial cells still form glandular structures but this formation is not proper.

In Fig. 2.2(c) and Fig. 2.2(d), regions marked with 2 are the examples of Grade 2 colon adenocarcinoma, where the distortion is moderate so that glands are moderately differentiated; here epithelial cells cannot form glandular structures but there are still some luminal centers that can be seen in the tissue. In Fig. 2.2(d), the region marked with 3 can be included in either side without affecting the medical interpretation since this region does not contain any epithelial cells, and thus, there is no colon adenocarcinoma associated with this region.

In Fig. 2.2(e) and Fig. 2.2(f), Region 2 is considered as Grade 3, where the

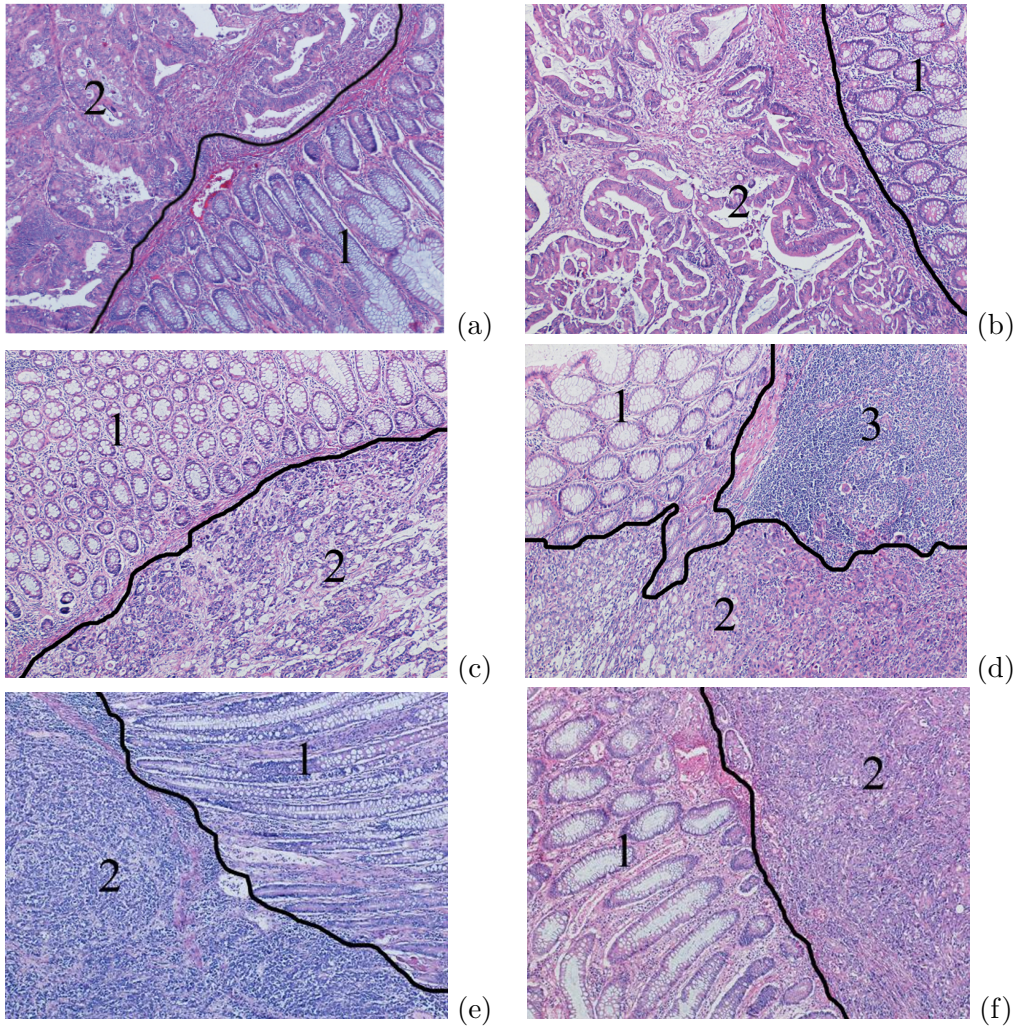


Figure 2.2: Example images of colon tissues stained with hematoxylin-and-eosin, which is routinely used to stain biopsies in hospitals.

distortion is high such that glands are only poorly differentiated; in this image, the epithelial cells are scattered all over the stromal area and there is no sign of a glandular structure or a luminal region.

If we focus on these microscopic images, we can identify the cytological components of a colon tissue, which are shown in Fig. 2.1. In the context of tissue image segmentation, the cytological components in the colon tissue and the spatial relations between these components are the primary concern. On the other hand, the exact localization of these components is quite difficult in our problem domain, which focuses on segmentation of low-magnification tissue images. Thus, we approximately locate the components making use of the domain specific knowledge. One of the main contributions of this thesis is to model the cytological components as primitive objects and extract information from these objects.

The aforementioned domain specific knowledge is based on several attributes of the cytological components. In tissues stained with the routinely used hematoxylin-and-eosin technique, the components appear in three main colors, namely, white, pink, and purple, and they usually form round structures. Hence, we formed our objects for all color types and in circular shapes. The object-oriented representation of a tissue image will approximately give information about the original component distribution of the tissue. More detail about this representation will be given in Chapter 3. Then we examine the distribution of the cytological components to detect the distortion caused by cancer by extracting spatial mathematical features from the distribution of the objects. This representation allows us to alleviate the effects of problems caused by pixel-level noise. Besides, it speeds up the algorithm since it summarizes the pixel domain into an object domain.

2.2 Related works on image segmentation

In image segmentation, the goal is to simplify and/or change the representation of an image into some model that is more meaningful and easier to analyze [51].

We may categorize the algorithms in literature mainly into two, according to the regions that they want to segment: object segmentation and scene segmentation. Object segmentation algorithms aim at extracting objects from an image, such as segmenting baseball players from the background in an image [52], extracting objects (e.g., books, shoes or pets) from an image of a room [53], and extracting glands from a histopathological image of a colon tissue [8].

On the other hand, scene segmentation algorithms mainly focus on extracting homogeneous regions from images such that each region has the same characteristics all over its pixels. For instance, some of these algorithms focus on segmenting natural scenes into separate regions of trees, sea, and sky [54, 55] and others focus on segmenting landmarks such as buildings, farms, roads and forests from satellite images [56, 57]. The problem stated in this thesis can be considered in the context of scene segmentation, since it is the segmentation of homogeneous regions (cancerous and normal regions) in histopathological images.

Many scene segmentation algorithms have been proposed thus far. We can group these algorithms according to two criteria: the information used for segmentation and the methodology used for segmentation. According to the information used, these algorithms can be generally grouped into two: those that use *intensity/color* or *texture* information of pixels.

The algorithms that use color information divide the image pixels into groups, usually based on their color histograms, by using different techniques, such as k-means clustering [58], fuzzy clustering [54, 59], watershed transformation [60], and thresholding [61, 62, 63]. For example, in [54], dominant colors of an image are computed in RGB color space and then pixels are clustered according to the dominant color distribution. In another study [64], HSV (hue-saturation-value) color space is used to cluster the pixels. There are also studies that apply clustering or thresholding to different color spaces or to different pairwise combinations of color channels to calculate multiple segmentation maps of the same image. They then fuse these intermediary maps to obtain the final segmentation [65, 66]. This kind of segmentation, which directly uses color information, gives promising results if the color distribution of different regions in an image is not similar.

However, if the color distribution of different regions is very similar as in the case of histopathological images, these approaches may not be sufficient.

In the latter case, the algorithms use texture information of pixels. For each particular pixel, they extract textural information considering the intensities of pixels located in the local neighborhood of this particular pixel. In order to differentiate different texture patterns, various types of texture descriptors are defined; such as Markov models [64], Laplacian of a Gaussian [67], and co-occurrence matrices [68]. However, in scene images, the texture within a homogeneous region (which is considered to have the same characteristics) could vary and this makes hard for a segmentation algorithm to estimate parameters for having the proper set of texture descriptors.

The approaches can be grouped according to the methodologies as well. $Region-based$ approaches group the image pixels into clusters, maintaining connectivity among the pixels of the same cluster. Examples include region growing algorithms [41, 69, 70], split and merge procedures [71, 72], edge-based segmentations [73], and watershed transformations [74]. The region growing usually works in a bottom-up fashion. In this methodology, pixels that are considered as homogeneous according to a criterion are selected as initial regions (seeds) and the remaining pixels are agglomerated to these initial regions with respect to the homogeneity criterion [75]. In split-and-merge methodology, the whole image is first considered as heterogeneous and it is continually divided into multiple partitions until each resultant partition is homogenous according to a criterion. Subsequently, these homogenous partitions are merged with their neighbors with respect to another homogeneity criterion [76]. In edge-based segmentation, pixels violating continuity are determined as the boundaries of segmented regions [68]. Finally, the algorithms that use watershed transformations, choose local minima of the gradient of the image as markers and start flooding from these markers. In watershed transformations over-segmentation is a common problem; thus these algorithms usually use region merging as their second step [77].

In the $region-based$ approach, the algorithms typically use color, color gradient, and/or texture to define the region homogeneity. They also combine color

and texture information. For example, in the JSEG algorithm [41], pixels are first quantized using color information, regions are then segmented using a homogeneity criterion that relies on the spatial distribution of these quantized pixels, and the over-segmented regions are finally merged based on their color information.

$Graph - based$ approaches use graphs in their segmentation methodologies. These approaches consider the image as a weighted graph where nodes represent pixels and the weight of each edge connecting two nodes represents the similarity between them. They then formulate image segmentation as a problem of partitioning this graph into components, minimizing a cost function. It has been proposed to solve this problem using different similarity measures, different cost functions, and different optimization methods [44, 64, 78, 79].

Finally, *statistical* approaches consider image segmentation as a probabilistic optimization problem. They model the image probability distributions directly, using parametric and non-parametric estimation [80, 81], or by using graphical models such as Markov random fields and Bayesian networks [82, 83].

Although all these approaches lead to promising results, the scene segmentation problem is not completely solved yet and there still remain different challenges to overcome for different applications. The main challenge lies in the nature of this problem. Image segmentation is closely related with human perception. Humans typically combine their background knowledge with image data to segment the image into its semantically uniform regions. To incorporate the human perception into segmentation, adaptive clustering algorithms have been proposed [54, 84]. These algorithms adaptively define color-texture descriptors that show spatial variations with respect to the image content. It has also been proposed to define descriptors at different scales for mimicking a human observer looking at the same image from different distances and to combine them in segmentation [85].

For the case of histopathological images, the aforementioned scene segmentation algorithms could yield misleading results due to non-ideal conditions in tissue preparation procedures including fixation, sectioning, and staining. The misconduct and variation in these procedures may produce a considerable amount of

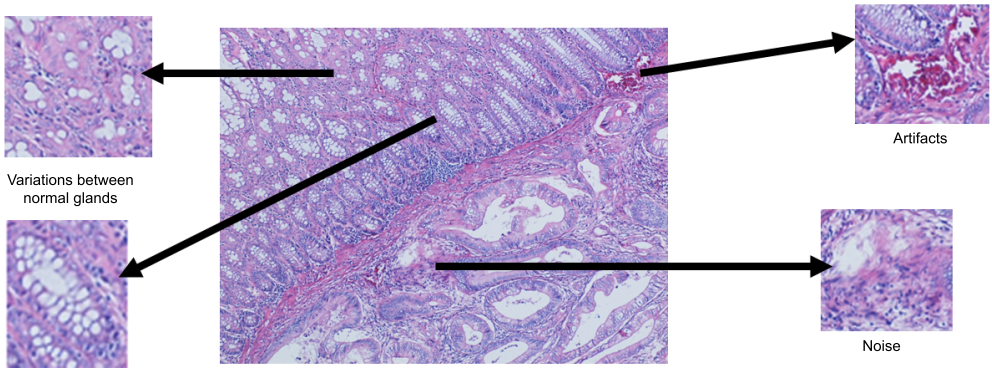


Figure 2.3: Examples of noise, artifacts, and variations caused by misconduct and variation in tissue preparation procedures.

noise, artifacts, and variation among different biopsy images. Some of these problems can be seen on an example image in Fig. 2.3. Furthermore, the hematoxylin-and-eosin technique, which is routinely used to stain biopsies in hospitals, results in similar color distributions in heterogeneous regions (see Fig. 2.3). Therefore, using only the color information of pixels in histopathological image segmentation is inefficient because of these similar distributions. Likewise, using the texture of pixels could give incorrect segmentations especially for the images that consist of noise, artifacts, and variation because of the non-ideal tissue preparation procedures.

Moreover, the histopathological image segmentation problem can be considered in two different types of scope. The first type is to locate biological objects such as cells and glands on an image [8, 86, 87, 88]. The second type, which is also the focus of this thesis, is to locate homogeneous regions in a heterogeneous image. In literature, there are only a few studies focusing on the latter problem. Most of these studies perform grid analysis, in which image segmentation is achieved dividing an image into fixed grids and classifying them in a supervised manner [9, 11]. In [11], Wang et al. take a whole biopsy slide as one huge image, divide it into grids, extract intensity-based and textural features from each grid entry, and classify these entries based on their extracted features. This approach could give non-robust results for images of tissues that are not properly prepared. Furthermore, it brings about the difficulty of the selection of a proper grid size and the need of high performance computing. In another

work [89], Kong et al. propose to classify pixels of a slide into four by using color-based and texture information. This approach is also facing with similar problems (noise and variations in tissues, improper parameter estimation, and high computational complexity). The use of low-level pixel information makes all of these approaches more vulnerable to noise and variations in histopathological images. Another group of studies consider pixels instead of grids [90, 91]. These studies characterize the pixels using features extracted within their neighborhood and process them to construct a segmented region. In characterizing a grid or a pixel, these studies use low-level color/texture descriptors that they define on image pixels.

All of the mentioned studies use pixel intensities and/or pixel textures to quantify the grids or pixels. However, they do not consider the background knowledge of the tissue organization to define their descriptors. Indeed, it is quite difficult to express this background knowledge in terms of pixels (i.e., by using pixel based feature descriptors). This is mainly due to the large variations observed in different biopsy samples and the noise occurs in preparing these samples and taking their images. The variations and noise typically cause local changes in pixel values. However, they do not change the semantics in the distribution of cytological tissue components on a large scale. For example, in Fig. 2.2, one could capture the normal appearance of a colon tissue in spite of the variations and noise observed in its normal regions. This is our main motivation behind defining new feature descriptors that better correlate with high-level image semantics.

There are also studies that use a supervised model to segment or classify the regions of histopathological images [92, 93, 94]. These studies aim to design specific mathematical features for different applications and build up a system that stores pre-known information for the segmentation process. However, the learning part of a supervised model requires using a dataset of images with their ground truths (gold standards) prepared by medical experts. Although such systems are desirable and known to be accurate with use of prior knowledge, they are not efficient systems in context of productivity. Therefore, an unsupervised segmentation algorithm may be preferred in many applications as a first step of an automated diagnosis system.

In order to improve the success of such unsupervised segmentation, raw pixel information should be preprocessed and transformed into a better model and the background knowledge should be incorporated in the segmentation algorithm. In this thesis, we propose new approaches for the effective and robust segmentation of histopathological tissue images. In the proposed algorithms, we introduce new texture measures that model the spatial distribution of cytological tissue components and the use of this texture measures in histopathological image segmentation.

Chapter 3

Object definition

The approaches proposed by this thesis rely on characterizing a tissue image with high-level texture features and using them in an efficient segmentation algorithm. To this end, we introduce *objseg features*, *graph run-length features*, and *object co-occurrence features* that quantify the spatial organization of tissue components. Before extracting these features, a tissue image is decomposed into a set of objects of different types, which approximately represent the tissue components. In this chapter, the decomposition process of the histopathological tissue images will be detailed. The decomposition process includes three main steps: clustering image pixels, preprocessing of the clustered pixels, and applying our circle-fitting procedure to extract circular objects from the clustered pixels.

3.1 Clustering image pixels

Tissue images used in our studies are in RGB color space. To cluster these pixels according to their color distributions, we use the k-means clustering algorithm over RGB values of the pixels. In this clustering, k is selected as three, because biopsies used in these studies are routinely stained with the hematoxylin-and-eosin technique in hospitals, which mainly gives white-like, pink-like, and purple-like pixels. For the initialization of the cluster centers in k-means, the principal

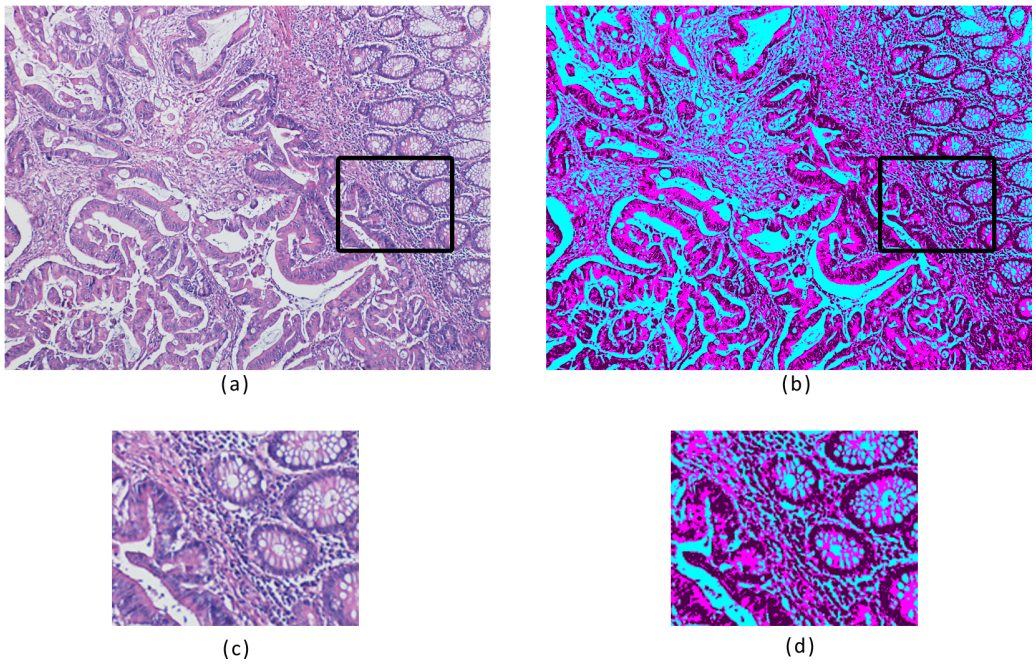


Figure 3.1: Results of the k-means algorithm on an example histopathological tissue image: (a) original image, (b) clustered pixels, (c) zoomed sub-image, and (d) clustered pixels of the zoomed sub-image. In the cluster maps, white-like, pink-like, and purple-like pixels are shown in cyan, pink, and purple, respectively.

component of the data is calculated, its range is divided into k equal intervals, and initial centers are defined as the averages of the data points falling in these intervals [95]. An example k-means clustering result of a tissue image is shown in Fig. 3.1. In this example, the image shown in Fig. 3.1(b) represents the clustered pixels of the original image shown in Fig. 3.1(a). In this figure, each cluster is shown with a different color; to easily relate the clustered pixels with original image pixels, white-like pixels in the original image are labeled with cyan in the clustered image, pink-like pixels are labeled with pink, and purple-like pixels are labeled with purple color. After obtaining it, the clustered pixel map is preprocessed to reduce noise and artifacts and then objects are located on the processed map by our circle-fitting procedure.

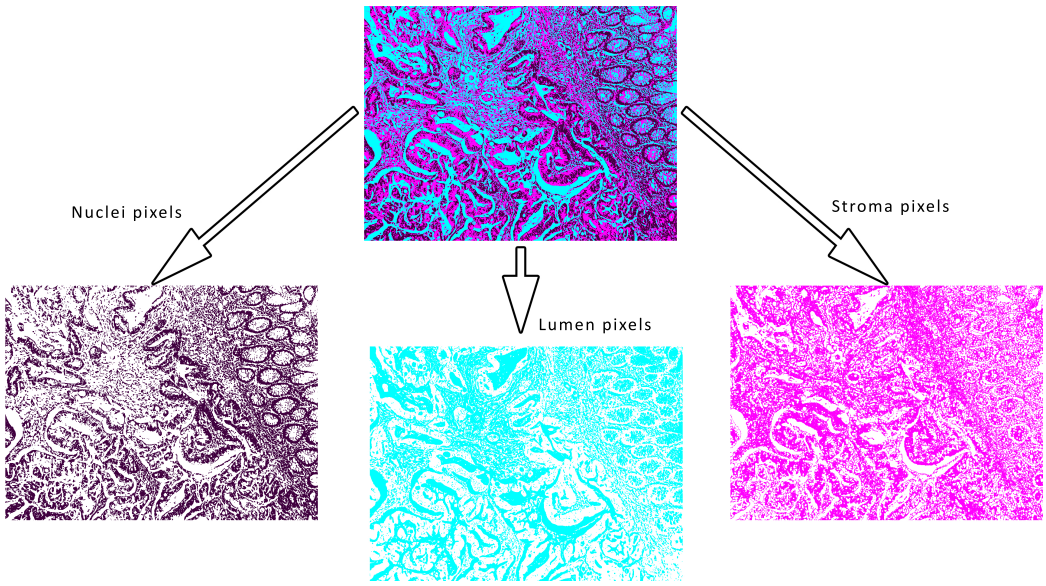


Figure 3.2: The separation of nuclei (purple-like), lumen (white-like), and stroma (pink-like) pixels in the preprocessing step.

3.2 Preprocessing

After clustering the image into three clusters, the preprocessing step uses morphological operators to remove noise from the clusters. This is done for smoothing the pixel components to prepare them for the circle-fitting procedure. In the first step of preprocessing, the clustered pixels are divided into three separate pixel maps: cluster 0 representing nuclei (purple-like) pixels, cluster 1 representing stroma (pink-like) pixels, and cluster 2 representing lumen (white-like) pixels. This separation is shown in Fig. 3.2. These maps are colored as purple, pink, and cyan respectively for the visualization.

The separated pixel maps are then dilated with a circular structural element of variable sizes. In our first study (ObjSEG), this structural element's radius was set to 2 since we want to extract a smaller number of objects with larger sizes for the sake of the computational performance of the application. To extract larger objects, ne need larger components on the clustered pixels, and thus, the raw pixel maps are dilated with structural elements of larger sizes. In our latter studies (GraphRLM and ObjCooc), we set this structural element's radius to 1

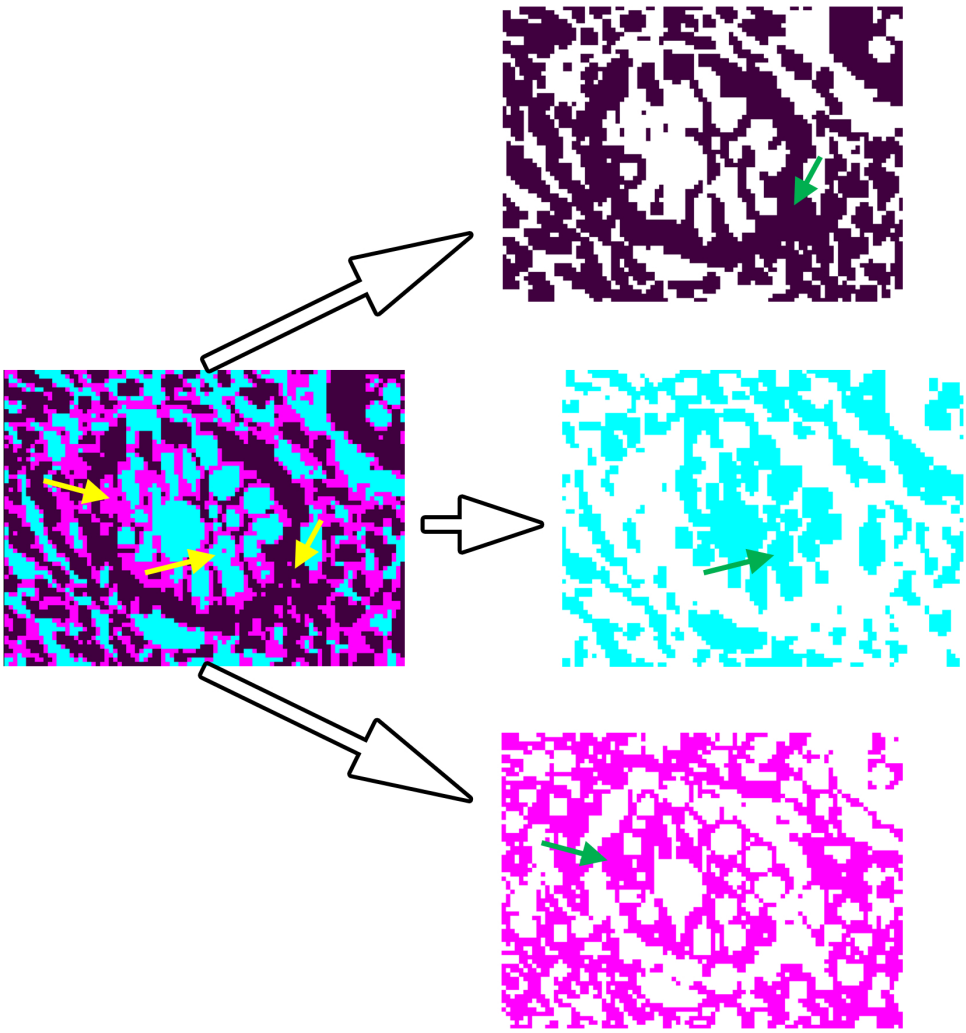


Figure 3.3: The effect of dilation operation on nuclei, lumen and stroma pixels in preprocessing step.

to generate more circles with smaller radii. When the pixels of one particular map are dilated, they grow over other maps' pixels. Hence, structural elements of smaller sizes are better to preserve the information in an original image.

The dilation process is effective to eliminate salt and pepper noise, which may prevent to locate relatively larger sizes of objects. After removing such noise, it is more likely to obtain objects that represent the tissue components better. Fig. 3.3 shows the benefits of using the dilation process. The noisy pixels shown with an arrow prevents defining a circle of a larger size on the local neighborhood of this pixel. This benefit can also be observed in an example result of the circle-fitting

procedure shown in Fig. 3.4.

After dilating a map, the pixels of the located objects are excluded from the remaining maps and these pixels are not used to define objects on the remaining maps. Therefore, the order in which the maps are dilated is very important. Since the distribution nuclei is major, we first dilate the pixel map of nuclei (purple-like) pixels. By doing so, we decrease the likelihood of losing information about nuclei. Then, we dilate the pixel map of lumen (white-like) pixels and finally the pixel map of stroma (pink-like) pixels.

3.3 Circle-fitting

The final step of locating objects on a tissue image is the circle-fitting procedure. In this step, the dilated cluster pixel maps are taken into account. First of all, circular masks with different radii are generated. Then, for each connected component of the cluster map, these circular masks are fitted in the descending order of their sizes. When a mask fits into a component, the pixels belong to this circle are marked and not used in next iterations. This continues until there are no circle masks left that are bigger than $minR$ and that fit into the connected components. The minimum radius $minR$ is a model parameter and can be set according to the minimum component size in the image or according to the purpose of the application. We use $minR = 9$ in the ObjSEG algorithm, since we wanted to obtain bigger objects for the sake of the computational performance. In the GraphRLM and ObjCooc algorithms, we set $minR = 5$ to obtain a larger number of objects to extract as much relationship information as we can get from the tissue components of a histopathological image.

On an example sub-image, the results of the circle-fitting procedure are shown in Fig. 3.4. The objects defined as nuclei, lumen, and stroma cluster maps are shown in Fig. 3.4(a), Fig. 3.4(b), and Fig. 3.4(c), respectively. The effect of the preprocessing operations can also be observed in these examples. After removing noisy pixels, the circle-fitting procedure can locate more representative circles.

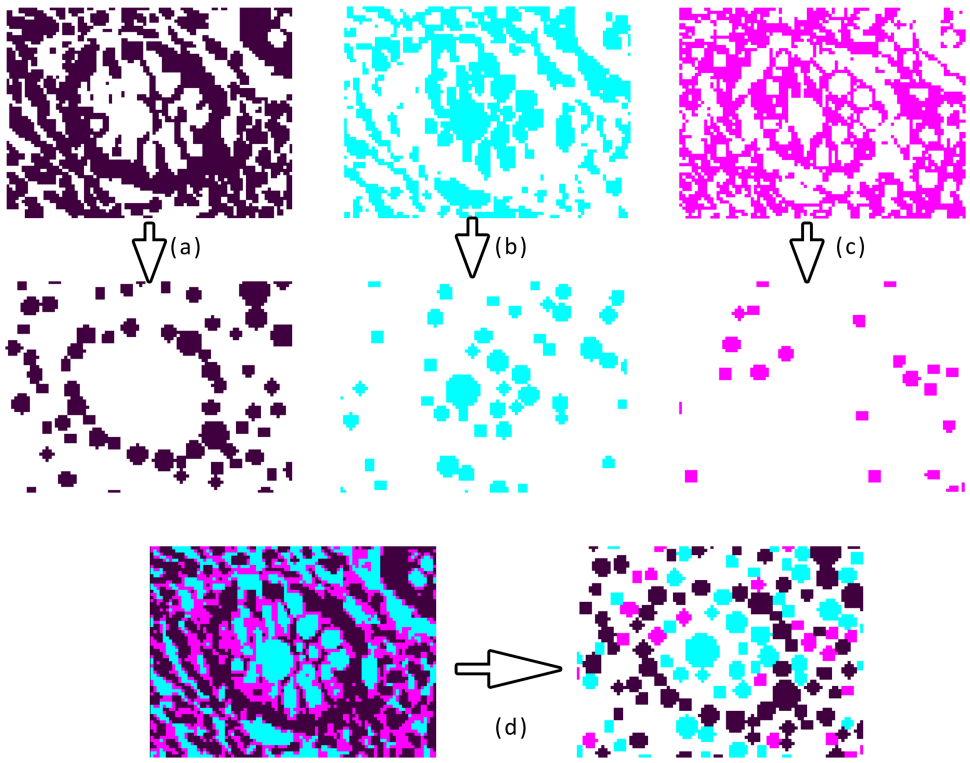


Figure 3.4: The results of the circle-fitting procedure on an example sub-image. The objects defined on nuclei, lumen, and stroma pixels are shown in (a), (b), and (c), respectively. All these objects are shown together in (d).

Fig. 3.4(d) visualizes all the objects defined together on a single image.

Fig. 3.5 shows the object decomposition after circle-fitting procedure on a larger image. As seen in this figure, even a person without any prior medical knowledge about tissues or cancer may distinguish between differently oriented regions by simply looking at the distribution of the extracted objects. This distinguishing feature of the object representation of the tissue components opened a new perspective for us. We defined several mathematical features on this representation and we proposed segmentation algorithms based on these features. These algorithms together with their features, methodologies, and experiments are given in the following chapters.

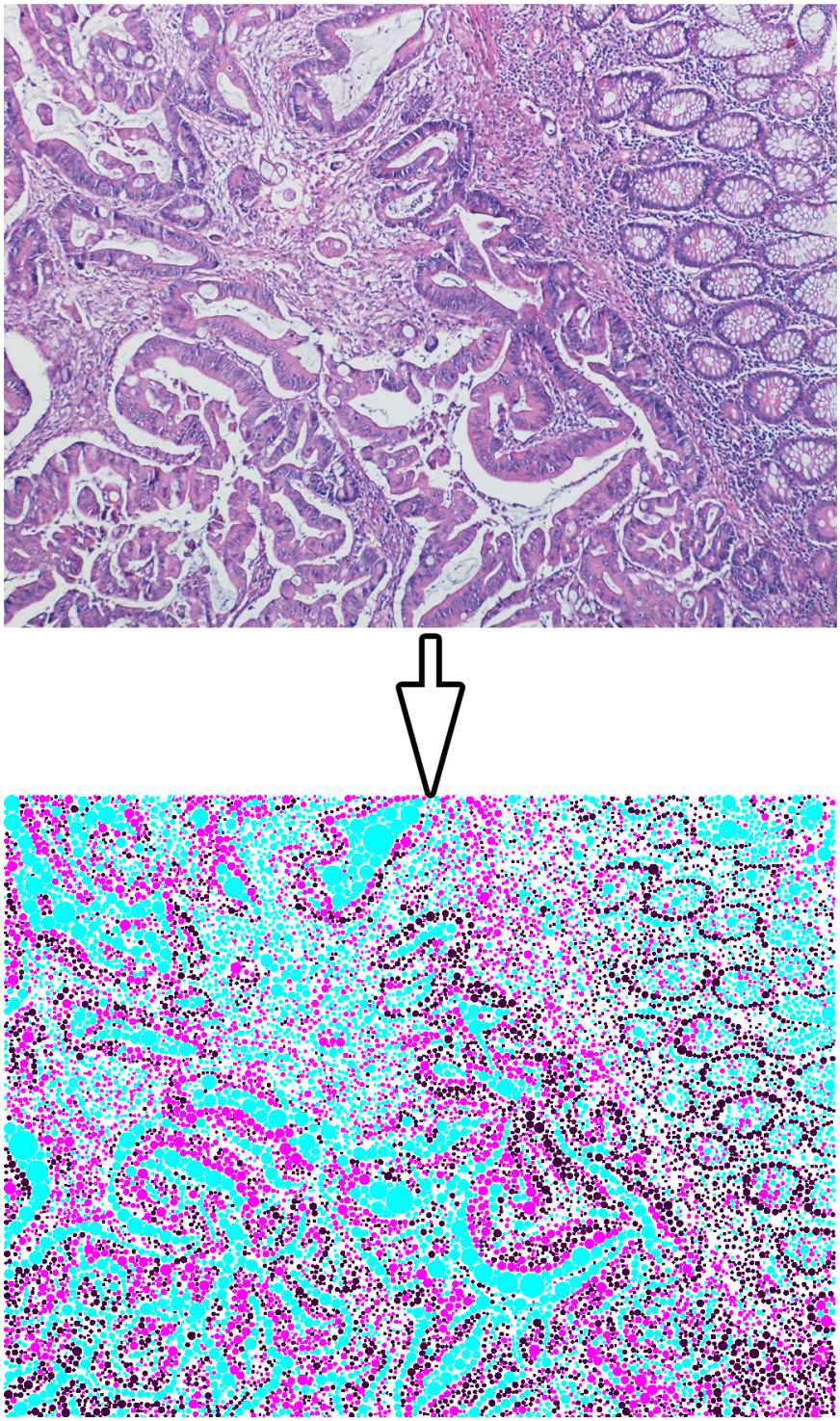


Figure 3.5: The results of the circle-fitting procedure on a larger image.

Chapter 4

Object-oriented texture analysis for unsupervised image segmentation (ObjSEG)

In a biopsy image, biologically different parts of the tissue are characterized with the spatial organizations of its cellular and connective tissue components. These organizations show differences, depending on the organ from which the tissue is taken. As mentioned before, this thesis focuses on colon biopsy images where the tissue components are organized to form glandular structures. In these images, epithelial cells are lined up around a lumen, forming a crypt (gland), and lymphoid cells take place in between these crypts. This organization deviates from its regular structure due to the existence of cancer. Furthermore, this deviation is aggravated with the increasing malignancy level. Thus, for the detection of cancer and its malignancy level, the regions containing such tumorous structures should be distinguished from those that contain the normal ones.

In this particular study, ObjSEG, we propose a segmentation algorithm that uses the fact that the structural organization of a tissue (i.e., the spatial distributions of cellular and connective tissue components with respect to each other) changes with the existence of cancer. To this end, we define our homogeneity

measure based on the texture of these components. We define primitive objects to represent these components (Chapter 3), instead of exactly identifying their locations, since this localization brings about a more difficult segmentation problem even for human experts.

4.1 Object-oriented texture definition

In our segmentation algorithm, we make use of the textural composition of the objects that we define for the representation of the tissue components. For this purpose, we propose a homogeneity criterion based on how the objects are distributed in size and in space. The homogeneity measures used in this study rely on the following observation: in a homogeneous region, for each particular object, there should be another one with the same type and the same size and that object should be on the symmetrically opposite side of the particular object with respect to the centroid of this region. To quantify this observation for every pixel, a window is located at the center of the pixel. In Chapter 3 we defined three types of objects. Here we divide each type into two according to their sizes to differentiate between large and small objects of all types. We use this separation because there is some prior information for our images such as, the observations that large nuclei typically belong to cancerous cells, large lumina usually corresponds to the center of luminal areas or unstained background regions, and small nuclei typically belong to stromal cells, which are found outside the glandular area. Hence, according to their sizes six different types of objects are defined in this study and 12 descriptors (two measures for each object type) are computed considering the objects falling in a specified window. The two measures defined for each object type are:

1. *Object size uniformity*: It measures the uniformity in the object size. For this reason, for each particular object type, we use the standard deviation of the areas of the objects that belong to this particular object type. If the objects are uniformly distributed in size, the standard deviation of their areas should be equal to zero. Note that here we use the coefficient of

variation since the different types of objects have areas of different scales.

2. *Object spatial distribution uniformity*: It quantifies how uniform the objects are distributed in space. For this purpose, for each particular object type, we compute the sum of the position vectors of every object, which belongs to this particular object type, with reference to the image centroid and use the magnitude of the resulting vector. If the objects are uniformly distributed in space, the magnitude should be equal to zero. We define this measure particularly for the objects rather than the pixels, as opposed to the previous work of Deng and Manjunath [41] where this measure is strictly defined on the individual pixels. For this measure, instead of pixels, using objects overcomes the problem of dealing with images that exhibit similar color distributions in their different regions, unlike the previous literature.

More technically, let $O = \{O_i^j | j = 1, 2 \dots 6; i \in N_j\}$ be the object set where O_i^j is the object with type j and id i and N_j is the number of objects with type j. Each object $O_i^j = \{X_i, Y_i, A_i\}$ is characterized with its centroids X_i and Y_i and its area A_i. Let W be the window located at a given pixel at X_w and Y_w, as shown in Fig. 4.1 on which the texture descriptors are calculated. As the first set of descriptors (object size uniformity), for each object type, the standard deviation Std^j of object areas are calculated as

$$Std^j = \sqrt{\frac{\sum_{k \ s.t. \ O_k^j \in W} (A_k - \mu^j)^2}{(n^j - 1)}} \tag{4.1}$$

where μ^j is the average area of the objects that fall in window W and has type j and n^j is the number of these objects.

As the second set of descriptors (object spatial distribution uniformity), for each object type, the sum Sum^j of the position vectors of the corresponding objects with respect to the window centroid (X_w,Y_w) is defined as

$$Sum^j = \left(\sum_{k \ s.t. \ O_k^j \in W} (X_k - X_w) \right) + \left(\sum_{k \ s.t. \ O_k^j \in W} (Y_k - Y_w) \right). \tag{4.2}$$

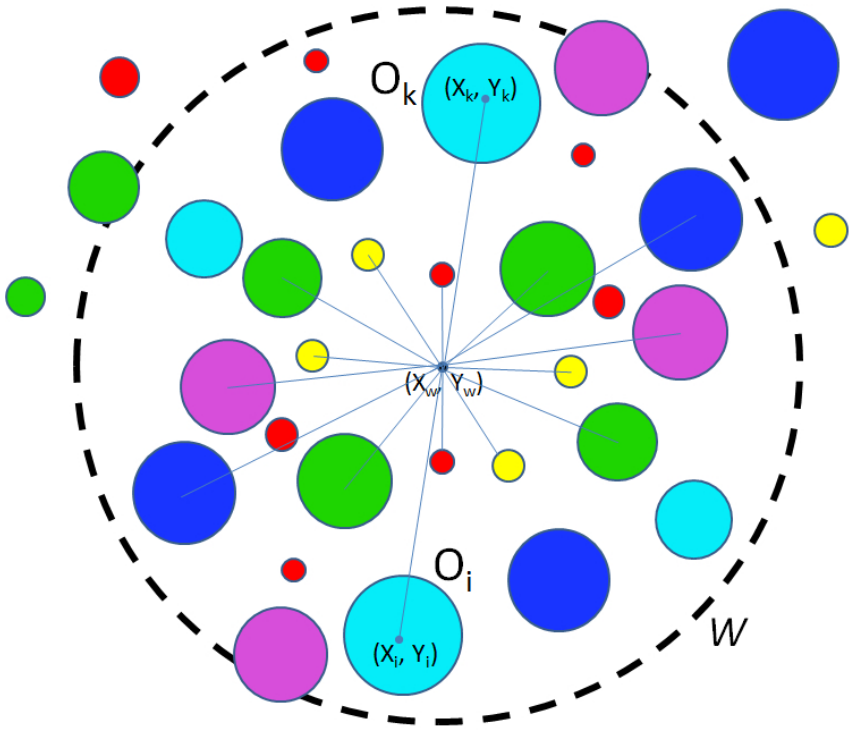


Figure 4.1: An example of a circular window W located at the centroid (X_w, Y_w).

4.2 Segmentation algorithm

In our segmentation algorithm, we propose to segment an image into regions for which the aforementioned two measures (object size uniformity and object spatial distribution uniformity) are minimized for all object types. As it is not feasible to minimize these measures for the entire image, we compute them over small windows and use them in our region-growing method. For this reason, we compute these measures using a window centered at each pixel in the image. Thus, for each pixel, there are a total of 12 uniformity measures, two measures for each of the six objects that we define for the representation of tissue components.

Our region-growing method is a three-step procedure. In the first step, we determine the initial seeds based on the uniformity measures of pixels. In the second step, we iteratively grow these initial seeds also based on the uniformity measures. In the third step, we merge the oversegmented regions employing the object distributions. Next, we explain these steps in detail; a schematic of the

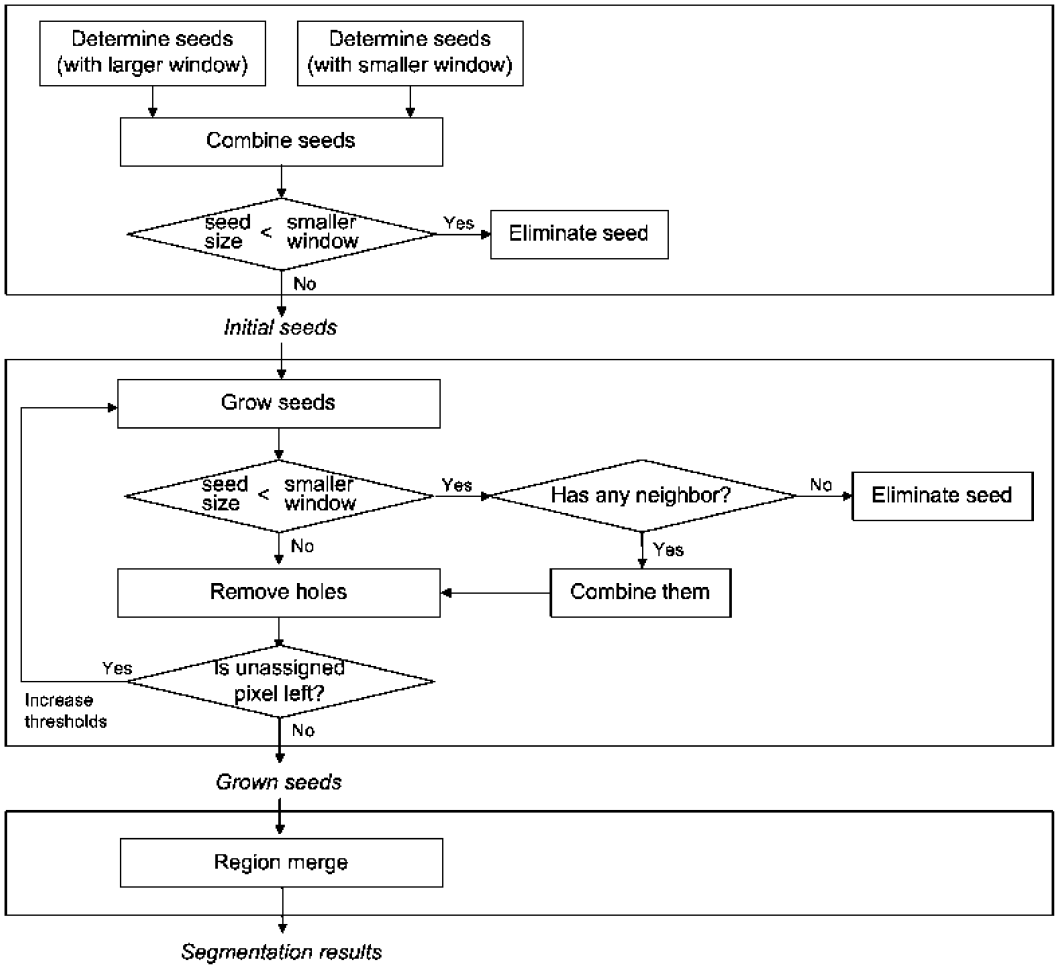


Figure 4.2: A schematic of the proposed region-growing method.

proposed region-growing method is provided in Fig. 4.2.

4.2.1 Seed-determination

We determine the seeds by connecting pixels for which all of the 12 uniformity measures are smaller than their corresponding thresholds. In our algorithm, for each measure j, we set a threshold $T_j = \mu_j + \sigma_j$, where μ_j and σ_j are the mean and the standard deviation of the associated measure computed over all pixels. Note that here we connect the pixels based on the four-connectivity.

In addition to determining the seeds based on our uniformity measures computed within a particular window $winL$, we repeat the same procedure using a smaller window $winS$, to capture finer textural information. Then, we combine the seeds obtained using both the larger and the smaller windows if their seeds do not overlap. In the case that these seeds overlap, we use the seeds obtained from the larger window and only the non-overlapping regions of the seeds obtained from the smaller window. At the end, we eliminate the seeds that are smaller than the size of the smaller window, whose area is denoted as $areaThr$. Fig. 4.3(c) shows the seeds computed for the image given in Fig. 4.3(a).

4.2.2 Seed-growing

In the next step, we start growing the initial seeds using the same set of 12 uniformity measures. In the beginning, we compute the threshold values by considering the pixels that are not assigned to any of the seeds in the seed-determination step. Using the four-connectivity, we connect the unassigned pixels for which all of the measures are smaller than their corresponding thresholds to form a new connected component. If this new component is a neighbor of a previously determined seed, we merge them together. In the case that this component has more than one neighbor, we merge it with its closest neighbor. If the component has no neighbors, then this component is considered as a new seed. We continue this iterative procedure for all of the remaining unassigned pixels

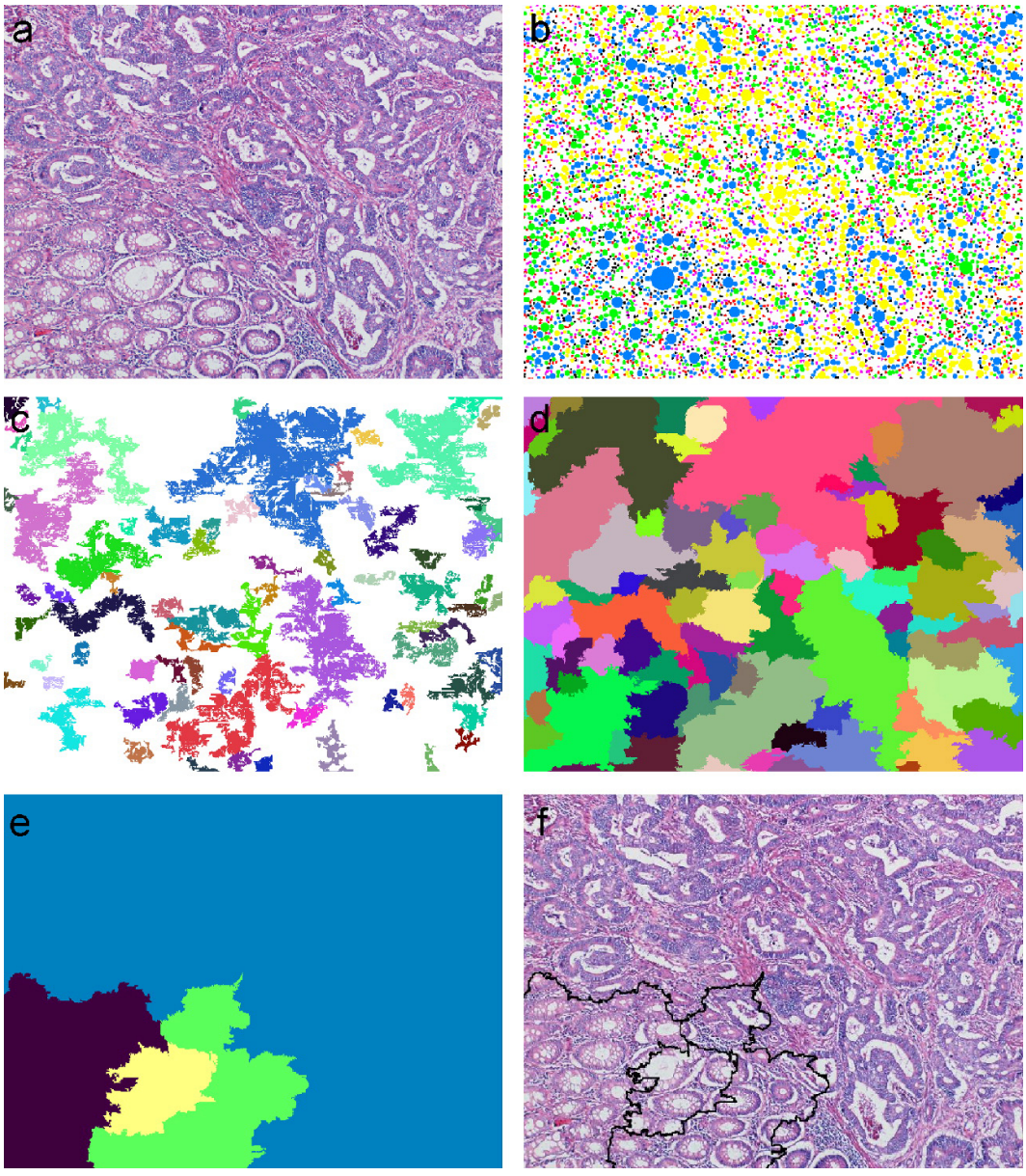


Figure 4.3: The steps for the object-oriented segmentation of a biopsy image: (a) start with the original tissue image; (b) locate the circular objects on the clustered image; (c) determine the seeds; (d) grow the seeds; (e) merge the regions; and (f) the segmented image.

until there is no unassigned pixel left with measures smaller than the thresholds. Note that, for each iteration, the threshold values are recomputed considering the mean and the standard deviation of only the remaining unassigned pixels from the previous iteration. After this iterative procedure is completed, if any of the determined seeds is smaller than the size of the smaller window, we connect it to its largest neighbor. In the case that it has no neighbors, we eliminate this seed. Finally, we fill in the holes of each seed.

We repeat this entire seed-growing procedure as necessary until there are no unassigned pixels left in the image. To do so, in its first iteration, we compute the threshold of the measures as $T_j = \mu_j + \sigma_j$ and update the threshold in the next iteration, by increasing the effect of σ_j incrementally each iteration.

For the image given in Fig. 4.3(a), the final regions computed at the end of the seed-growing step are shown in Fig. 4.3(d).

4.2.3 Region-merge

The seed-growing step often results in oversegmented regions. For this reason, in the final step, we merge these regions based on their object distributions. In our region-merge step, we use two different criteria to characterize a region: (1) the percentage of the total area of the same type of objects in the region and (2) the percentage of the combined areas of the different objects that correspond to the same cluster in the region. Using these criteria, we first merge a region with its closest neighbor if its size is smaller than the larger window size. In the case that its size is larger, we only combine it with its closest neighbor only if the distance between them is smaller than the merge threshold $mergeThr$. It is worth noting that after merging a region with its neighbor, we update the criteria accordingly and continue the merge with the updated criteria. For the image in Fig. 4.3(a), the regions obtained at the end of the region-merge step are illustrated in Fig. 4.3(e) and the final boundaries are superimposed on the original image in Fig. 4.3(f).

4.3 Experiments

In our experiments, we use the microscopic images of colon biopsy samples of 16 randomly chosen patients from the Pathology Department archives in Hacettepe School of Medicine. Each biopsy sample consists of a 5-6 micron thick tissue section stained with the hematoxylin-and-eosin technique.

The images of these samples are taken in the RGB color space using a Nikon Coolscope Digital Microscope; 5× microscope objective lens is used and the image resolution is 1920 × 2560. Each colon biopsy image contains both normal and cancerous (colon adenocarcinomatous) regions. Cancerous regions contain tumors with different grades. In Figs. 4.4–4.7, the first image of each row consists of the manual segmentation provided by our medical expert. For each tissue image, the regions are labeled as either cancerous or normal and the boundary between these regions are drawn in red. In between these boundaries, there are also regions that can be included in either side without affecting the medical interpretation; such regions are shaded in red.

In Figs. 4.4–4.7, the second column of each row shows the segmentation results for each image that are obtained by our object-oriented segmentation algorithm. To compare our results with those obtained from a pixel-based segmentation algorithm, we also run the JSEG algorithm, which is proposed by Deng and Manjunath [41]. This algorithm separately uses the color information of pixels and their texture composition to segment an image into its homogeneous regions. In our experiments, we use the program provided by its authors in their web site. The results obtained by this program are also presented in the third column of Figs. 4.4–4.7. For both our segmentation algorithm and the JSEG algorithm, the selection of the merge threshold parameter affects the results. In Figs. 4.4–4.7, we provide the results of each of these algorithms, selecting the best merge parameter for each image. Additionally, for the JSEG algorithm, we also select the best color quantization threshold and the best number of scales for each image.

In our algorithm, besides the merge parameter, we use the same set of parameters for all images. To define our circular primitives, we set the area threshold

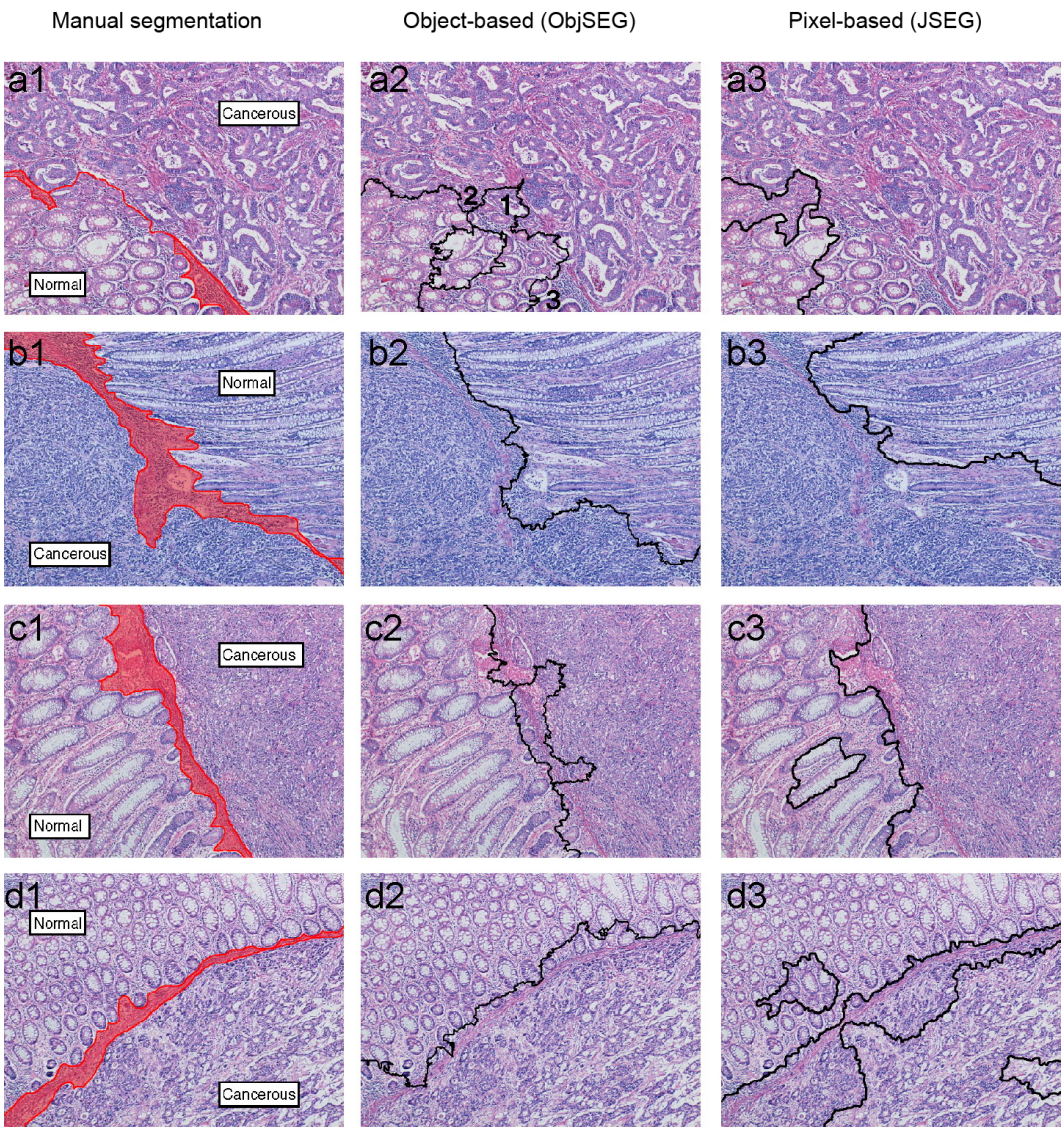


Figure 4.4: The segmentation results. (a1)-(d1) The manual segmentations; (a2)-(d2) the results obtained by our object-based algorithm; (a3)-(d3) the results obtained by the pixel-based JSEG algorithm.

parameter to be 100 pixels so that it is large enough to eliminate the noise and small enough to represent the tissue components. Also in defining the objects from the circular primitives, we use an object area threshold of 150 pixels to accommodate different tissue components represented by the same cluster. In the seed-determination, we use the window sizes of $winL = 257$ and $winS = 65$ pixels for the large and small windows to determine sufficiently coarse and fine textures in our images with a resolution of 1920×2560. In the region-growing step, we increment the effect of the standard deviation by 10 percent of its value in each iteration for computational reasons.

For the colon biopsy image shown in Fig. 4.4(a), our object-based algorithm leads to accurate segmentation. It results in only one cancerous crypt (marked with 1) being included in the normal region and only a small amount of normal crypts (marked with 2 and 3) being included to the cancerous region (Fig. 4.4(a2)). On the other hand, the pixel-based algorithm yields a segmented region for cancerous parts that contain a significant amount of normal crypts (Fig. 4.4(a3)). Similarly, for the image given in Fig. 4.4(b), our object-based segmentation algorithm yields accurate results and greatly improves the specificity and the accuracy compared to the pixel-based algorithm. Higher specificity (i.e., having less number of false positives) is important for correct grading of cancer. In cancer grading, the grade measures how much a tissue differentiates from the normal. The correct grading of cancer is critical since it affects the selection of the treatment and is an important predictor for the survival time of a patient. In colon adenocarcinoma grading, the differentiation is quantified according to how much the glands of a cancerous region are similar to the normal gland. False positives affect this quantification and cause to select a lower grade.

For the images given in Fig. 4.4(c) and (d), both the pixel-based and the object-based algorithms yield accurate results in the segmentation of their adenocarcinomatous and normal regions.

For the images given in Figs. 4.5(e)-(g), both of the algorithms result in over-segmentation. However, although the object-based segmentation algorithm does not have a heterogeneous region, which includes both cancerous and normal parts,

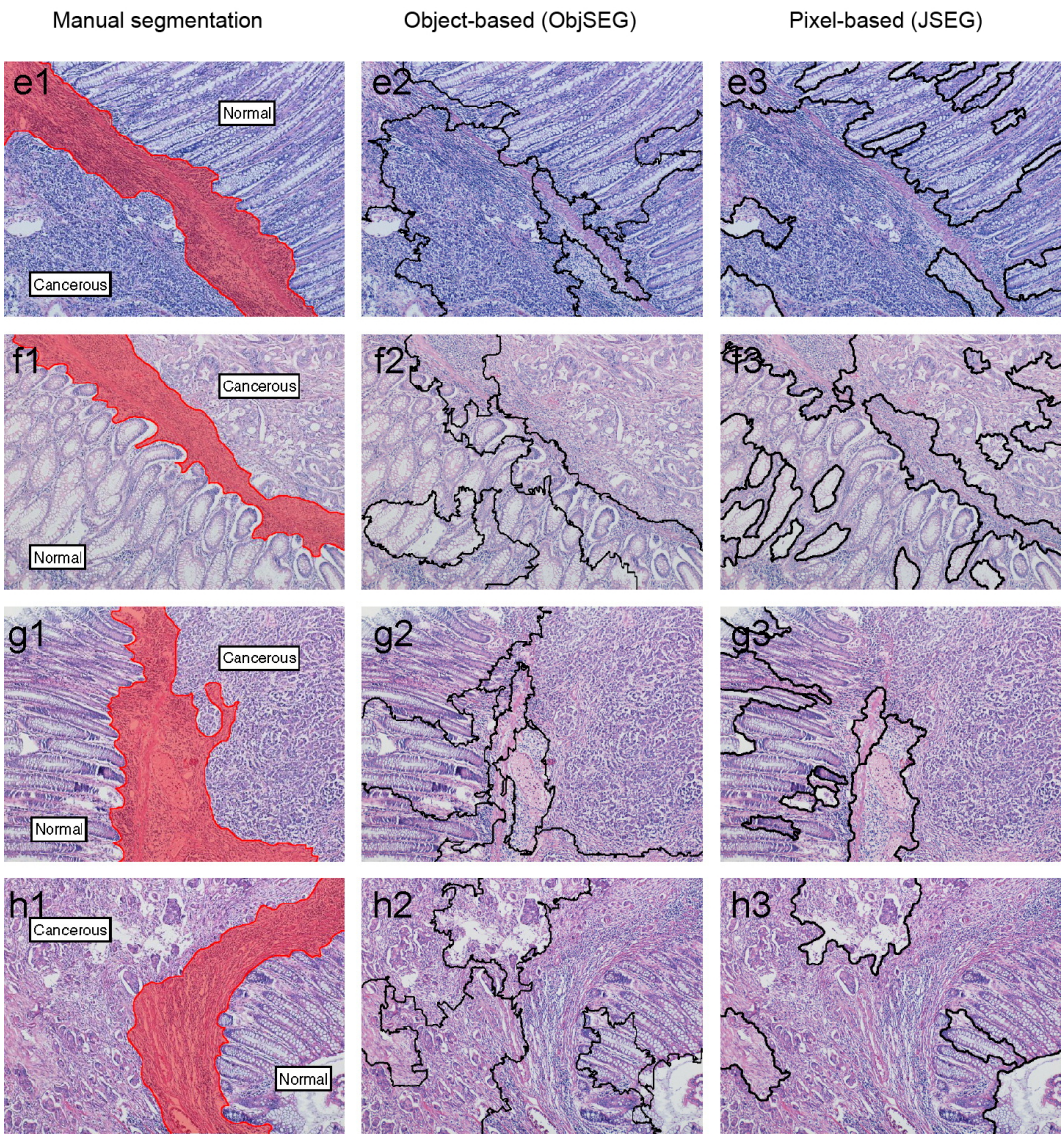


Figure 4.5: The segmentation results. (e1)-(h1) The manual segmentations; (e2)-(h2) the results obtained by our object-based algorithm; (e3)-(h3) the results obtained by the pixel-based JSEG algorithm.

the pixel-based segmentation algorithm has one that contains both of these parts. Similarly, for Fig. 4.5 and Fig. 4.6, the pixel-based algorithm leads to heterogeneous regions, and thus, yields lower segmentation accuracy compared to the object-based algorithm.

In Figs. 4.6(j) and (k), there are heterogeneity in the tumor such that it contains regions of both grade 2 (marked with 1) and grade 3 (cancerous regions other than that marked with 1). Besides, in Fig. 4.6(j), in the regions marked with 2, the tumor infiltrates into normal regions. For this image, our object-based algorithm successfully segments the normal region that is located in the upper left from the cancerous parts and captures the heterogeneity in cancerous regions. On the other hand, although the pixel-based algorithm identifies some part of this normal region, it could not capture the heterogeneity in the tumor. For this image, neither the object-based nor the pixel-based algorithms can segment the normal crypts that are in the top right of the image. For the image given in Fig. 4.6(k), both the pixel-based and the object-based algorithms capture the heterogeneity in the tumor. For the image shown in Fig. 4.6(l), both of the algorithms segment the normal and cancerous regions with an acceptable accuracy. At a first glance, it could be considered that the pixel-based algorithm yields better results since the object-based algorithm results in oversegmentation. However, when the normal regions are examined carefully, it is observed that normal regions contain some non-tumorous transformations (oncocytic and hyperplastic transformations) as well as lymphoid aggregations (regions marked with 1 and 2, respectively). Although the pixel-based algorithm does not capture these transformations, the object-based algorithm is successful to do that and distinguish the regions containing such transformations from those that do not contain any. The identification of such heterogeneities is important for cancer grading and prognosis. The heterogeneity in the tumor is known to affect the cancer grade, while non-cancerous tissue transformations are presently believed to be the prognostic parameters, which is further expected to be better understood in the future.

For the image shown in Fig. 4.7(m), the object-based algorithm incorrectly segments a normal region (marked with 1 in Fig. 4.7(m2)) and oversegments the

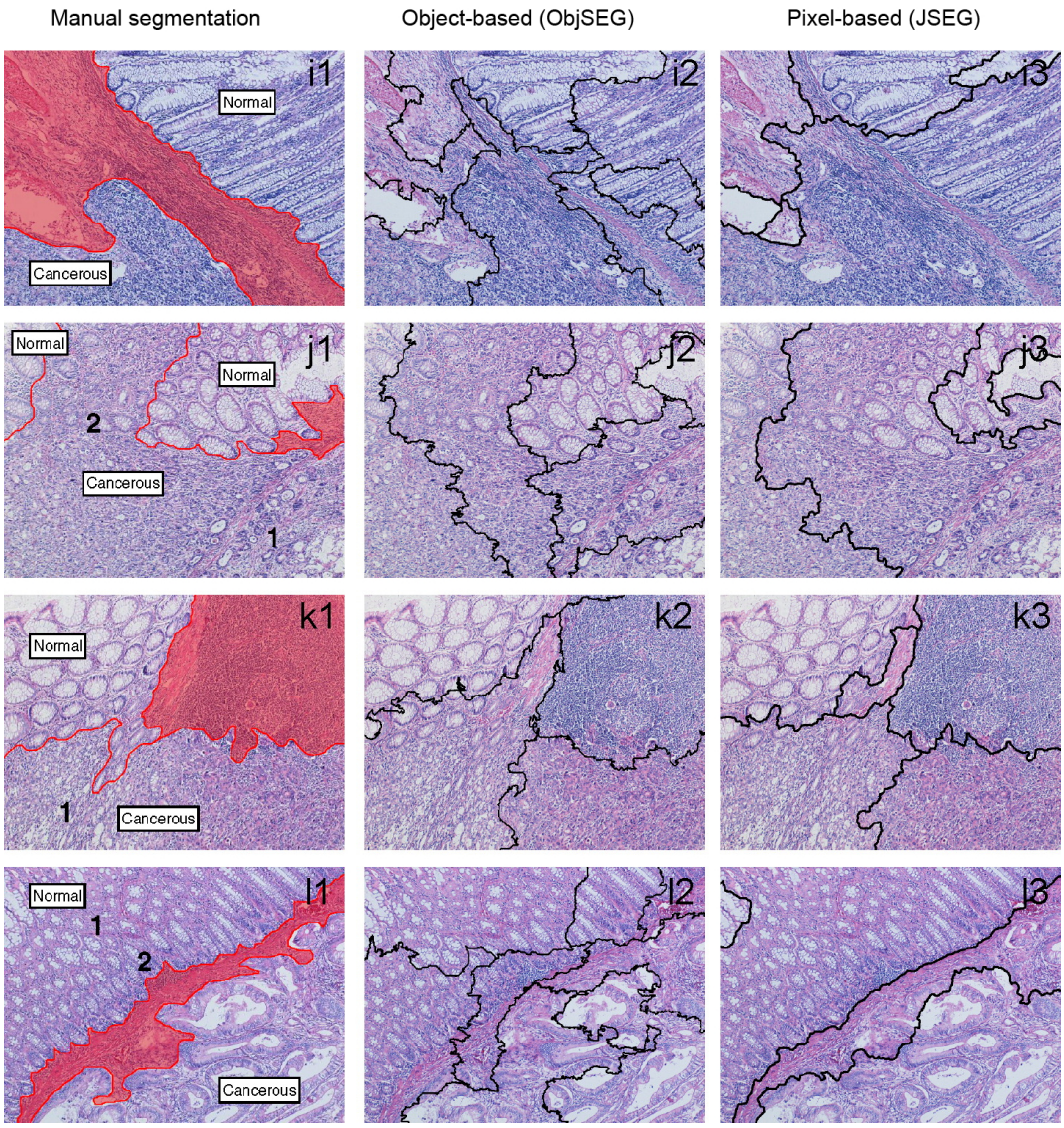


Figure 4.6: The segmentation results. (i1)-(l1) The manual segmentations; (i2)-(l2) the results obtained by our object-based algorithm; (i3)-(l3) the results obtained by the pixel-based JSEG algorithm.

remaining cancerous regions. For this image, the pixel-based approach achieves a successful segmentation.

For the image given in Fig. 4.7(n), both the pixel-based and the object-based algorithms lead to a heterogeneous region, which contains both normal and cancerous parts (regions marked with 1 in Figs. 4.7(n2) and (n3)). Besides this heterogeneous region, our segmentation algorithm results in more segmented regions compared to the pixel-based algorithm; this is attributed to the prominent lymphoid reaction in the tumor. Similarly, for the images given in Figs. 4.7(o) and (p), both of the algorithms result in heterogeneous regions (regions marked with 1 and 2 in Figs. 4.7(o2), (o3), (p2), and (p3)). In spite of these heterogeneous regions, both of the algorithm yield acceptable accuracies for all these images.

In order to quantitatively compare the segmentation results of the object-based and the pixel-based algorithms, we calculate the true positive, false positive, true negative, and false negative rates using the manual segmentation as the gold standard and then report the sensitivity, specificity, and accuracy of these algorithms. Since both our object-oriented algorithm and the JSEG algorithm are unsupervised segmentation methods, they do not output the class of the segmented regions. Therefore, we consider the class of the dominant region as the label of the region and calculate the true positive, false positive, true negative, and false negative rates accordingly. In our calculation, we do not consider the pixels that could be included in either side (either cancerous or normal region) without affecting the medical interpretation (the pixels that are shaded in red in Figs. 4.4–4.7). We report the average and the standard deviation of the sensitivity, specificity, and accuracy percentages in Table 4.1. This table shows that our object-oriented algorithm yields higher sensitivity and specificity percentages and that it improves the accuracy of the JSEG algorithm for the segmentation of histopathological images. To investigate whether or not this improvement is significant, we use the Wilcoxon test with a significance level of 0.05. This test exhibits that this improvement is statistically significant.

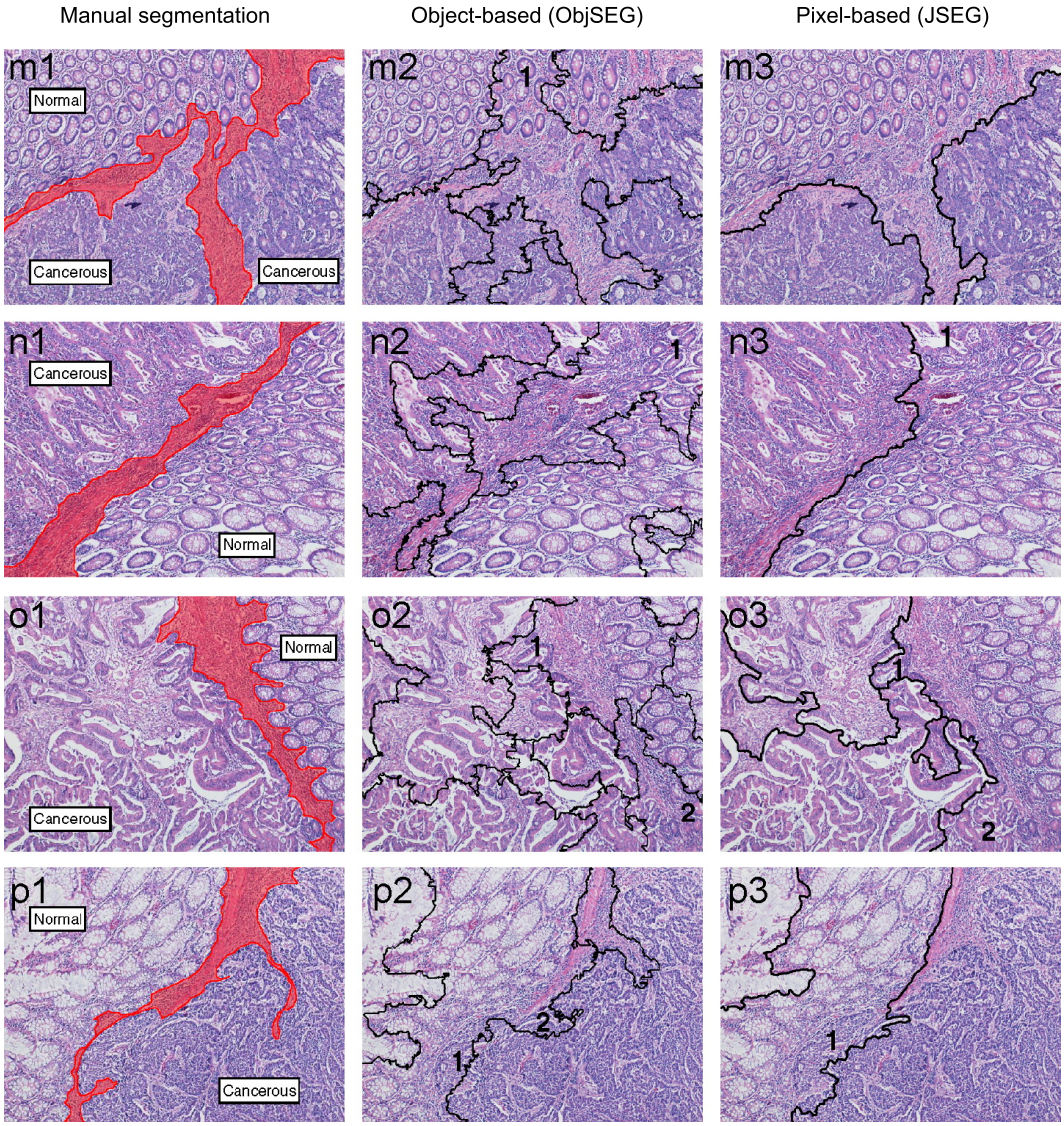


Figure 4.7: The segmentation results. (m1)-(p1) The manual segmentations; (m2)-(p2) the results obtained by our object-based algorithm; (m3)-(p3) the results obtained by the pixel-based JSEG algorithm.

Table 4.1: The average and the standard deviation of the sensitivity, specificity, and accuracy percentages for the proposed object-based and the pixel-based algorithms.

	Object-based	Pixel-based
Sensitivity	96.05 ± 5.72	86.25 ± 31.32
Specificity	92.22 ± 10.56	78.27 ± 26.38
Accuracy	94.89 ± 3.77	86.78 ± 11.46

4.4 Modified region growing algorithm for ObjSEG

Although ObjSEG improves the results of its pixel-based counterparts, it has a problem of finding a common parameter set ($mergeThr$ parameter) that works for all image instances, which reduces its robustness. This study extends the previous work to alleviate this problem and increase the robustness of the segmentation. In this current study, we propose a new region growing algorithm, in which the growing process depends on object-to-object relationships, instead of pixel connectivity [48]. It is different than the ObjSEG algorithm, which grows the initially segmented regions on pixel connectivity. Our experiments show that the use of object-to-object relationships in region growing increases the segmentation performance. It also improves the robustness of the algorithm, enabling to select a common parameter set for all image instances that leads to good segmentation results. The illustration of the region growing algorithm is given in Fig. 4.8. In the following subsections, we provide the details of this algorithm.

4.4.1 Object-to-object relationship definition

First, we need to define object-to-object relationships. To do so, we construct a Delaunay triangulation on the object centroids. Then, any two objects are said to be adjacent if they share an edge in the constructed Delaunay triangulation.

For extracting homogeneity measures, the same computations are done as explained in Section 4.1. We again use six types of objects grouped according to their clusters and sizes and define the ObjSEG features on these objects.

To remind this feature definition, let $O = \{O_i^j | j = 1, 2 \dots 6; i \in N_j\}$ be the object set where O_i^j is the object with type j and id i and N_j is the number of objects with type j. Each object $O_i^j = \{X_i, Y_i, A_i\}$ is characterized with its centroids (X_i, Y_i) and its area A_i. Here we locate a window W at the centroid of each object O_i^j as shown in Fig. 4.9, and calculate the texture descriptors in this

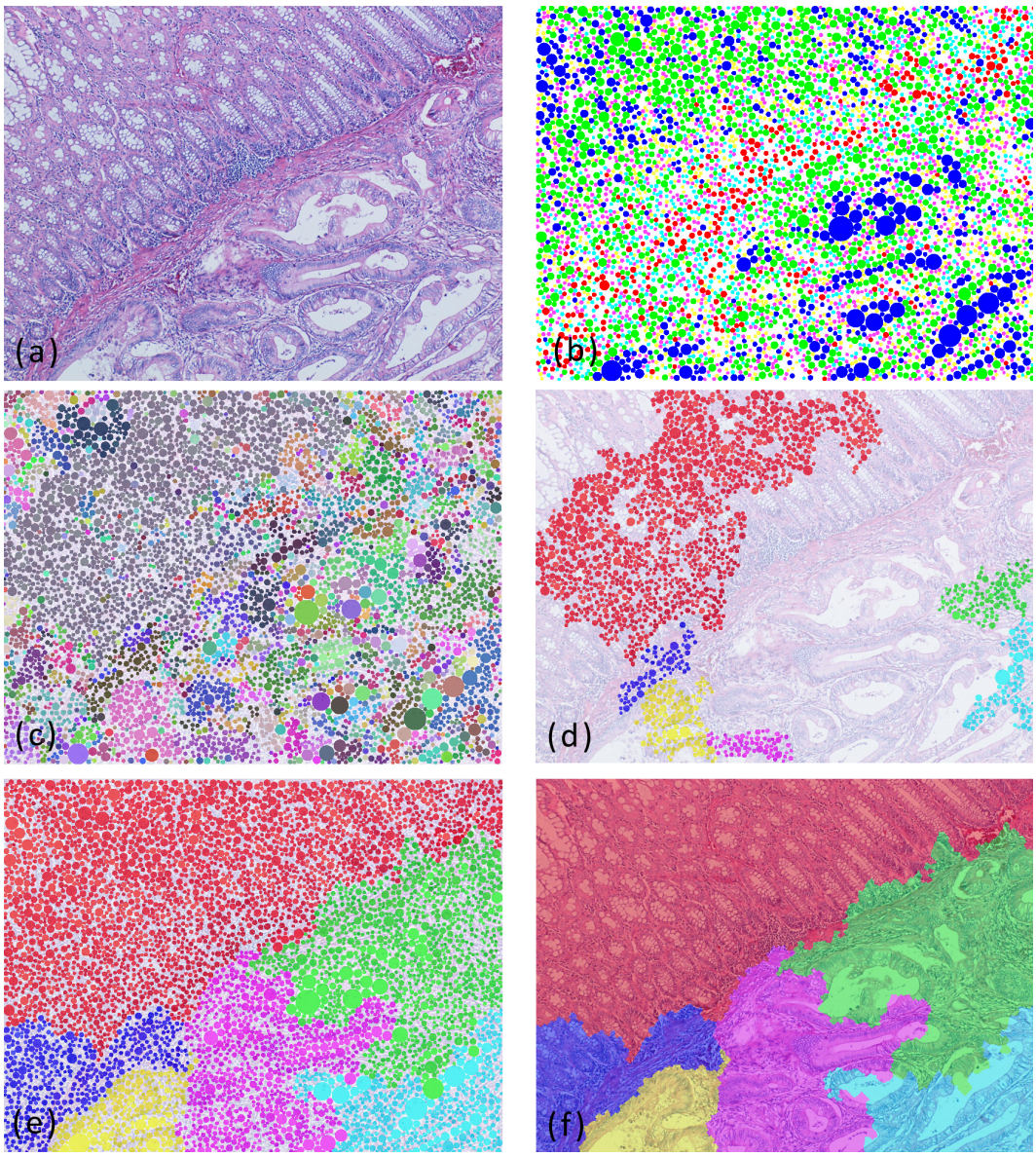


Figure 4.8: The illustration of the proposed region growing algorithm: (a) the original image, (b) circular objects that approximately represent the cytological tissue components, (c) all of the object groups, (d) seed groups after eliminating the small-sized object groups, (e) grown regions, and (f) final boundaries of the grown regions.

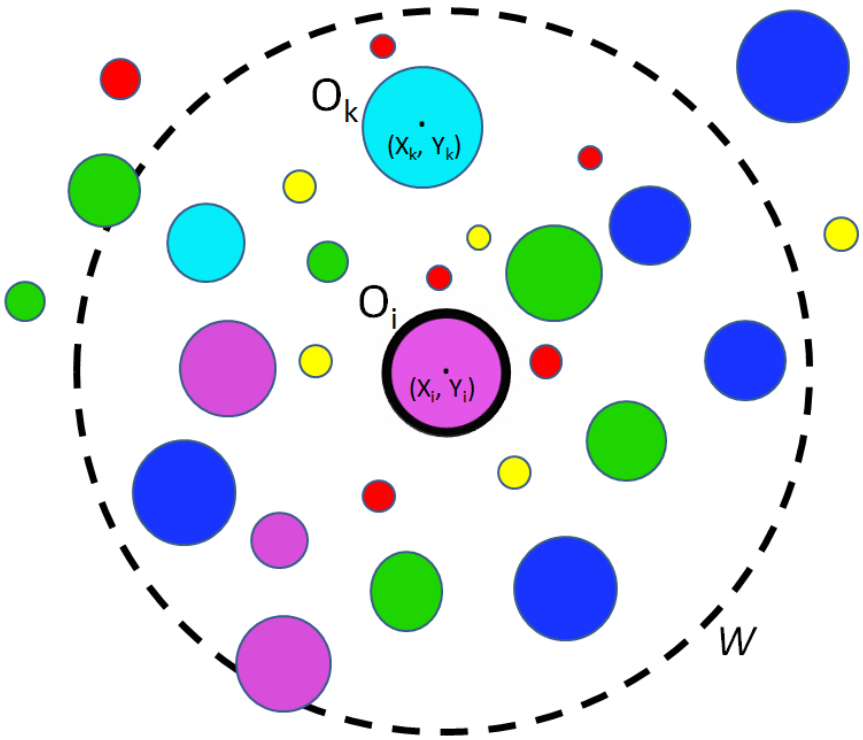


Figure 4.9: An example of a circular window W located at object centroid (X_i, Y_i).

window. Therefore, the window centroid denoted by (X_w, Y_w) in the previous section corresponds to the object centroid (X_i, Y_i). Likewise, we compute the Std^j and Sum^j measures for each object O_i^j as defined in Equations 4.1 and 4.2 respectively.

Since we only use objects in the region growing process, these measures are computed only for objects by locating the window at the centroid of each object. In addition, since we have already calculated the object relations, it is possible to iterate over the objects without using any pixel information. This approach increases the accuracy of the algorithm while decreasing computation time time so that the algorithm works faster than the ObjSEG algorithm.

4.4.2 Region growing algorithm

In the first step of the region growing algorithm, seed groups are identified based on the similarity of adjacent objects. Then, similar objects are grouped together such that the Euclidean distance between any pair of the adjacent objects in this group is below a similarity threshold. Finally, the large-sized groups that contain objects more than an object threshold are considered as seeds. Fig. 4.8(c) shows the objects of the same group with the same color. Fig. 4.8(d) shows the seeds that are obtained by eliminating the small-sized groups.

In its second step, seeds are iteratively grown by appending the remaining objects to one of the seeds. For doing this, an individual remaining object is appended to an adjacent seed group if the distance between this object and the seed group is smaller than the similarity threshold that is relaxed by its 10 percent in every iteration. The descriptors of a seed group are calculated by averaging the descriptors of all objects that belong to this seed group. When all objects are assigned to a seed group, the algorithm constructs the Voronoi diagram on the objects to find the final boundaries of the grown regions. The grown regions and their final boundaries are illustrated in Fig. 4.8(e) and Fig. 4.8(f), respectively.

4.4.3 Experiments

We conduct our experiments on 16 randomly chosen colon tissue images that contain both normal and cancerous regions. The tissues are stained with hematoxylin-and-eosin and their images are captured using a Nikon Coolscope Digital Microscope with 5× microscope objective lens. The images are taken in the RGB color space and their resolution is 1920×2560.

The proposed algorithm has three parameters: window size ($winSize$), similarity threshold ($simThr$), and object threshold ($objThr$). To select the parameter set, we use leave-one-out cross validation; for each particular image, we determine the parameter set on all other images excluding this particular image and obtain its test segmentation result. For that, we consider all possible combinations of the following sets $winSize = \{32, 64, 96, 128\}$, $simThr = \{0.25, 0.50...3.00, 3.50, 4.0\}$, and $objThr = \{10, 25, 50... 100, 150... 250\}$ and select the one that leads to the best performance over all images except the excluded one. In defining the best performance, we consider both the accuracy and the number of segmented regions: we select the parameter set that leads to the best accuracy and that gives at most 10 segmented regions. Note that if only the accuracy was considered, we would select the parameter set that leads to very high accuracies but at the same time very high number of regions. Table 4.2 reports the average quantitative test results and the average number of segmented regions. The quantitative results are calculated comparing the segmented regions with the manual segmentation provided by our medical collaborator. The evaluation methodology is the same as the one used in $ObjSEG$. We consider the class of the dominant region as the label of the region and calculate the true positive, false positive, true negative, and false negative rates accordingly. In our calculation, we do not consider the pixels that could be included in either side (either cancerous or normal region) without affecting the medical interpretation (the pixels that are shaded in black in Figs. 4.10 and 4.11).

In order to understand the effectiveness of the proposed algorithm, we compare its results with our previously proposed $ObjSEG$ algorithm [45], which uses a similar set of homogeneity criteria but a different region growing procedure in its

Table 4.2: The average and the standard deviation of the sensitivity, specificity, and accuracy percentages for the object-based and the pixel-based growing algorithms.

	Object-based growing	Pixel-based growing (ObjSEG)
Accuracy	86.5 ± 11.1	82.8 ± 12.5
Sensitivity	86.0 ± 26.5	92.1 ± 16.3
Specificity	82.9 ± 23.4	66.1 ± 32.8
No. Regions	5.9 ± 2.2	6.1 ± 2.2

segmentation. $ObjSEG$ has also model parameters: small and large window sizes ($winS$ and $winL$), area threshold ($area$), and merge threshold ($mergeThr$). We select its parameter set also using leave-one-out cross validation, considering the following candidate sets: $winS = \{32, 64\}$, $winL = \{128, 256\}$, $area = \{5000, 7500... 20000, 25000... 50000\}$, and $mergeThr = \{0.0, 1.0, 1.5, 2.0...4.0\}$. The quantitative test results obtained by the $ObjSEG$ algorithm are also given in Table 4.2.

Figs. 4.10 and. 4.11 show the visual results of the proposed algorithm in (a1)-(h1) and those of the $ObjSEG$ algorithm in (a2)-(h2) on six example images. On these images, segmented regions obtained by each algorithm are shown in different colors. The images also include the boundaries of cancerous and normal regions that are manually drawn by our pathologist collaborator; regions that can be included in either side without affecting the medical interpretation are shaded in black.

The quantitative and visual results show that the proposed region growing algorithm gives better segmentation performances. However, there are still errors in some images such as the one shown in Fig. 4.10(f1). This is due to using a common parameter set; for such images, better results can be achieved with different sets. Nevertheless, the quantitative results demonstrate that the proposed algorithm yields better accuracy, sensitivity, and specificity values when a common parameter set is used for all images. Note that the quantitative results reported for $ObjSEG$ in previous section are the ones that are obtained by separately optimizing the merge threshold for each image. Moreover, the proposed algorithm leads to a reasonable number of segmented regions even though it does

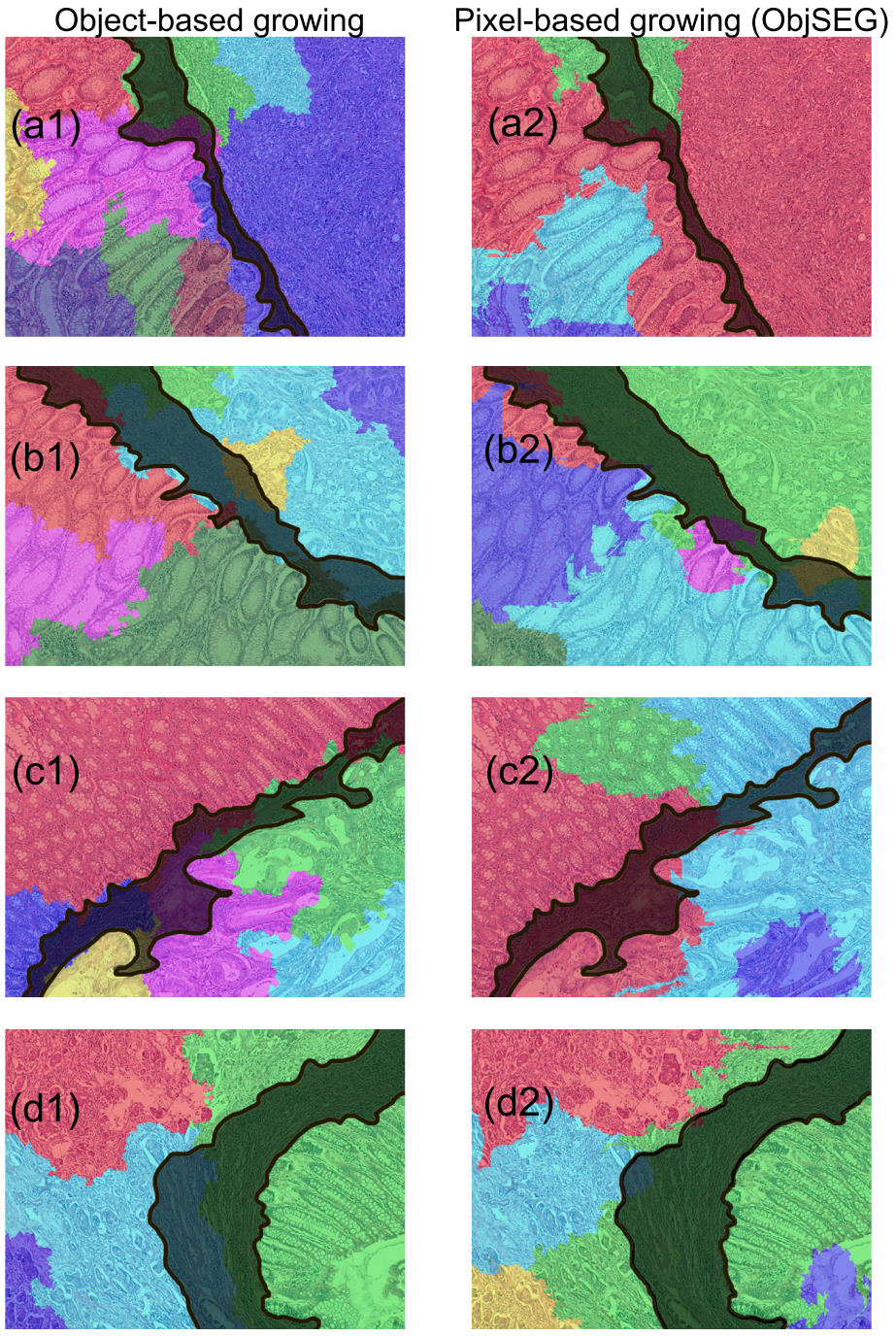


Figure 4.10: The visual results of the object-based growing segmentation algorithm (a1)–(d1) and the pixel-based growing (ObjSEG) algorithm (a2)–(d2). Segmented regions are shown with different colors. The manual segmentations are also indicated in these images.

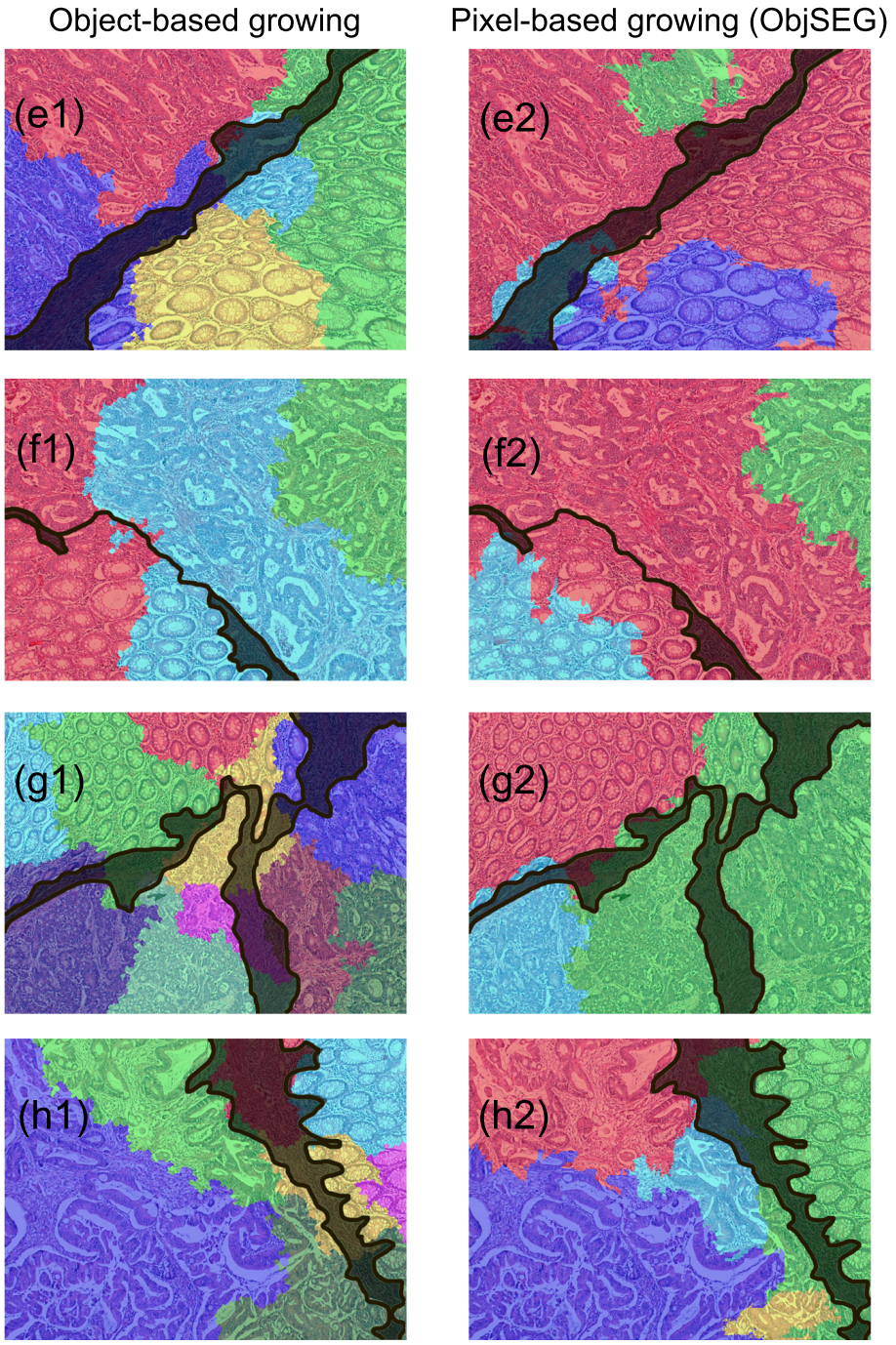


Figure 4.11: The visual results of the object-based growing segmentation algorithm (e1)–(h1) and the pixel-based growing (ObjSEG) algorithm (e2)–(h2). Segmented regions are shown with different colors. The manual segmentations are also indicated in these images.

not have an explicit region merge step.

4.4.4 Summary

In this particular study, we proposed a new homogeneity measure based on the distribution of the objects. For this purpose, we defined objects to represent tissue components including epithelial tissue components, connective tissue components, and luminal structures. Using this object-oriented measure, we demonstrated a new object-based segmentation algorithm. As opposed to the existing algorithms that use pixel-based information in defining their homogeneity measure, our segmentation algorithm uses object-based information, for the first time.

Working with colon biopsy images with similar color distributions in their heterogeneous regions, we demonstrate that our object-oriented algorithm significantly improves the accuracy in segmenting tumorous regions and also other non-cancerous tissue transformations compared to its pixel-based counterpart.

Following this study, we introduced a new region growing algorithm for the unsupervised segmentation of tissue images. This new algorithm relies on using the similarity of objects that approximately represent cytological tissue components. Working with the images of colon tissues, our experiments show that the proposed region growing algorithm leads to better results compared to the previous algorithm that uses similar criteria but a different region growing procedure.

This improvement showed that using relational objects is more effective than computing only the features on the image pixels. For this reason, we use the neighborhood information of the objects by forming an object based graph for our next studies.

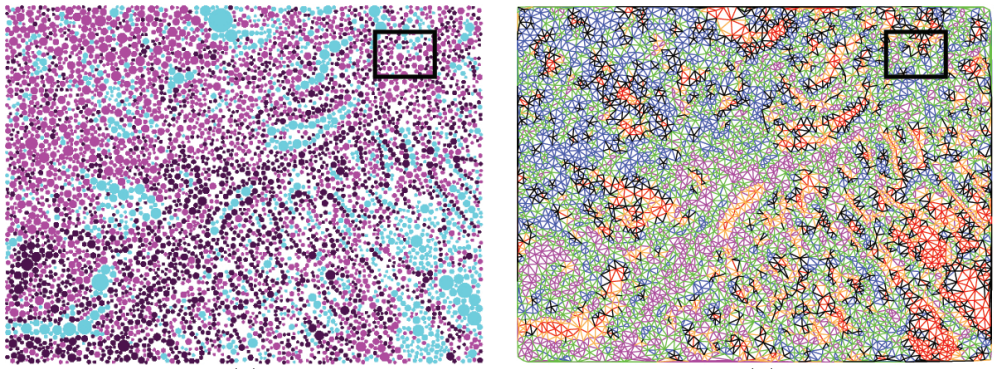
Chapter 5

Graph run-length matrices for histopathological image segmentation (GraphRLM)

The GraphRLM algorithm also relies on modeling the spatial distribution of cytological components within a tissue. For this purpose, it introduces a new texture measure to quantify the spatial relations of these components. This texture definition first generates a graph on the cytological components, then defines a run-length matrix using the edges of the generated graph, and finally extracts a set of texture features from the graph run-length matrix. The details of these steps are given in Sections 5.1–5.3. The segmentation algorithm that uses this new texture definition is explained in Section 5.4.

5.1 Graph generation

In this work, we represent the spatial relations between cytological tissue components using the color graphs, which are defined for the classification of histopathological images in our research group [3]. For the construction of these color graphs, the cytological tissue components are approximately represented



(a) (b)

Figure 5.1: A graph generated for representing the spatial distribution of cytological components within a tissue: (a) circular primitives representing the tissue components and (b) labeled edges defined in between these primitives.

with three different types of primitives, each of which is defined on the pixels of one of the three prominent colors observed in a tissue image. These colors are white, pink, and purple and their corresponding pixels are obtained by k-means clustering. As explained in Chapter 3, we propose to approximately represent tissue components with circular primitives since their exact localization gives a much more difficult segmentation problem. Specifically, lumina and epithelial cell cytoplasms are represented with white primitives, stroma are represented with pink primitives, and cell nuclei are represented with purple primitives. For the image given in Fig. 2.2(a), these primitive are shown in Fig. 5.1(a).

After the primitives are identified, a color graph is generated by constructing a Delaunay triangulation on the centroids of these primitives and then labeling each triangle edge according to the primitive types of its end points. As there are three primitive types in an image, its graph could consist of six different edge types. In Fig. 5.1(b), the edges assigned in between the primitives are shown; here, edges of different types are illustrated with different colors. For better illustration, Fig. 5.2 shows an enlarged picture of graph nodes and edges for the subimage that is confined within a rectangle in Fig. 5.1.

In our work, although we use a similar graph generation algorithm with [3], the way and aim of using these graphs are completely different. In [3], graph theoretical features (such as average degree and diameter) are used to classify

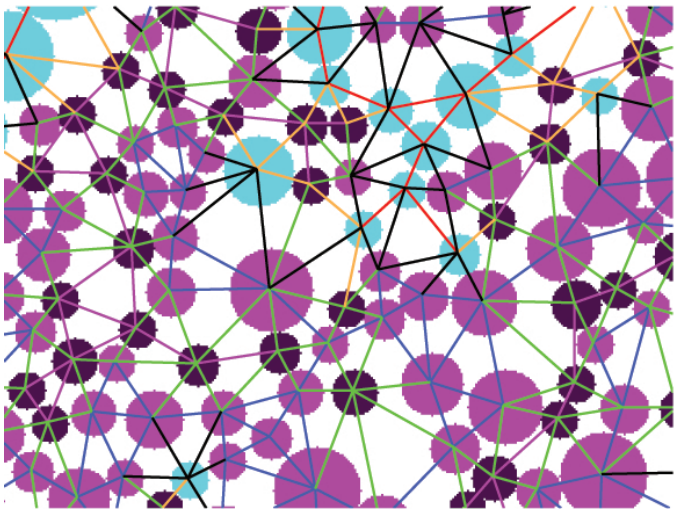


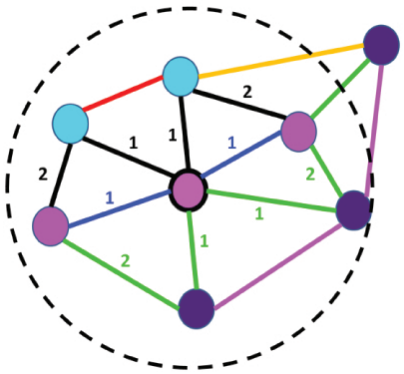
Figure 5.2: The graph of the subimage confined within a rectangle in Fig. 5.1.

histopathological images that are completely homogeneous. On the other hand, in this current study, we propose to employ the graph edges to define a texture measure that is used for the segmentation of heterogeneous histopathological images.

5.2 Run-length matrix calculation

After obtaining a graph, we calculate the run-length matrix of this graph to quantify the spatial relations of cytological tissue components (i.e., the texture of components). In defining graph run-length matrices, we make use of the idea of calculating gray-level run-length matrices. On a gray-level image, the run-length matrix I quantifies the coarseness of a texture in a specific direction. Given a direction, $I(i, j)$ is the number of runs of pixels with a gray-level i and a run-length j. A gray-level run is defined as a set of consecutive pixels with the same gray value in the given direction [49].

Our proposed approach uses graph-edge runs instead of using gray-level runs. It defines a graph-edge run as a path that starts from an initial node and contains nodes all of which are reachable with a set of edges of the same type. Given the



type\run	1	2	3
red	0	0	0
pink	0	0	0
black	0	**2**	0
blue	**2**	0	0
green	0	**2**	0
orange	0	0	0

Figure 5.3: Illustration of calculating a graph run-length matrix for a single initial node, which is shown as a thick bordered pink circle.

initial node, the graph run-length matrix entry $R(c, l)$ is the number of graph-edge runs with an edge type c and a path length l. As the graphs are undirected and unweighted, the length of a path is defined as the number of hops required to reach from the initial node to the furthermost node in the path.

In calculating the graph run-length matrix of a single initial node, the algorithm first locates a circular window at the center of this initial node. Then, for each particular edge type, it extracts paths by employing the breadth-first search algorithm on edges that are of this particular type and that are located within the circular window. Figure 5.3 illustrates the calculation of a graph run-length matrix for a single node that is shown as a thick bordered pink circle. To calculate the graph run-length matrix of an entire region, the algorithm accumulates the run-length matrices of the nodes located in this region.

5.3 Feature extraction

From the original definition of a gray-level run-length matrix, Galloway [49] proposes to define five different texture features whose definitions are given in Table 5.1. In this table, n_r is the total number of runs in the run-length matrix ($\sum_i \sum_j I(i, j)$) and p is the number of pixels in the image. Our proposed approach takes the original definitions of the first four features and modifies them for the graph run-length matrices to define its texture features: short path emphasis,

Table 5.1: Texture features for gray-level run-length matrices.

$$\text{Short run emphasis} = \frac{1}{n_r}\sum_i\sum_j I(i,j)/j^2$$

$$\text{Long run emphasis} = \frac{1}{n_r}\sum_i\sum_j I(i,j)\times j^2$$

$$\text{Gray level nonuniformity} = \frac{1}{n_r}\sum_i\left(\sum_j I(i,j)\right)^2$$

$$\text{Run-length nonuniformity} = \frac{1}{n_r}\sum_j\left(\sum_i I(i,j)\right)^2$$

$$\text{Run percentage} = \frac{\sum_i\sum_j I(i,j)}{p}$$

long path emphasis, edge type nonuniformity, and path length nonuniformity.

The *short path emphasis* gives more importance to shorter graph-edge runs than the longer ones, dividing the number of runs by the square of their lengths. First, this feature, SPE, is calculated regardless of the edge types (similar to Table 5.1). Then, it, $SPE(c)$, is also calculated for each of the six edge types separately, considering only the runs of the corresponding type. The equations of these descriptors are given as follows. In these equations, n_r is the total number of runs in the graph run-length matrix ($\sum_c\sum_l R(c,l)$) and $n_r(c)$ is the total number of runs corresponding to edge type c.

$$\text{SPE} = \frac{1}{n_r}\sum_c\sum_l R(c,l)/l^2 \tag{5.1}$$

$$\text{SPE(c)} = \frac{1}{n_r(c)}\sum_l R(c,l)/l^2 \tag{5.2}$$

The *long path emphasis* gives higher weight to longer graph-edge runs than the shorter ones, multiplying the number of runs by the square of their lengths. Likewise, this feature, LPE, is calculated regardless of the edge types as well as it, $LPE(c)$, is calculated for each of the six edge types separately. The equations

of these descriptors are given as follows.

$$\text{LPE} = \frac{1}{n_r} \sum_c \sum_l R(c,l) \times l^2 \tag{5.3}$$

$$\text{LPE(c)} = \frac{1}{n_r(c)} \sum_l R(c,l) \times l^2 \tag{5.4}$$

The *edge type nonuniformity* determines how the distribution of edge types affects the texture. It takes its lowest value when the runs are evenly distributed over all edge types. Similarly, the *path length nonuniformity* determines how the distribution of path lengths affects the texture. It takes its lowest values when the runs are evenly distributed over all path lengths. In the following equations, the edge type nonuniformity, ETN, and the path length nonuniformity, PLN, are defined as follows.

$$\text{ETN} = \frac{1}{n_r} \sum_c \left(\sum_l R(c,l) \right)^2 \tag{5.5}$$

$$\text{PLN} = \frac{1}{n_r} \sum_l \left(\sum_c R(c,l) \right)^2 \tag{5.6}$$

5.4 Segmentation algorithm

The proposed approach employs a region growing algorithm that uses the graph run-length features for segmentation. In this algorithm, region growing is achieved on the primitives, not on the pixels as in the case of previous studies [41, 45]. For each primitive, a window is centered at the centroid of the primitive and a run-length matrix is accumulated over the matrices of the primitives that are located in this window, as explained in Section 5.2. The graph run-length features calculated on this accumulated matrix are used as the descriptors of the primitive, which is located at the center of the window. In the subsequent steps of the algorithm, these descriptors are used in (dis)similarity calculation. In our algorithm, Euclidean distance is used as a dissimilarity measure.

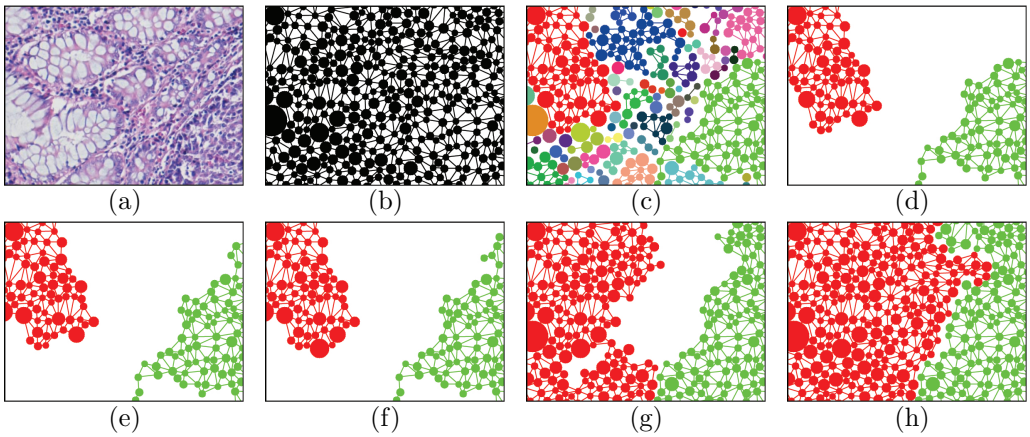


Figure 5.4: The illustration of the segmentation algorithm: (a) an original subimage, (b) its constructed graph [no color information is presented for better visualizing the subsequent steps], (c) graph connected components obtained after disconnecting dissimilar primitives [primitives and edges of the same component are shown with the same color], (d) initial seeds obtained after eliminating small-sized components, (e) grown seeds after one iteration, (f) grown seeds after two iterations, (g) grown seeds after 15 iterations, and (h) final grown seeds.

In the seed determination step, seed regions are found using the neighborhood relations defined by the constructed graph. To this end, the distance between every pair of adjacent primitives is computed and a pair is disconnected if the distance between them is over *a distance threshold*. Then, the small-sized connected components that include less number of primitives than *a component size threshold* are eliminated and the remaining components are considered as the initial seeds. Figs. 5.4(b)–5.4(d) illustrate the steps of seed determination for a small subimage shown in Fig. 5.4(a); here primitives and edges of the same component are shown with the same color.

In the region growing step, remaining primitives are iteratively assigned to the initial seed regions. In each iteration, primitives that are adjacent to at least one of the seed connected components are considered. A primitive is assigned to its closest seed if the distance between them is less than a grow threshold. Here, we start the grow threshold with the distance threshold, which is used in the seed determination step, and increase it by its 10 percent in each iteration. Region growing continues until there are no unassigned primitives left. For a seed component, the run-length features are obtained averaging them over all of

its primitives. Figs. 5.4(e)- 5.4(h) show the grown seeds obtained at the end of different iterations.

In the region merge step, adjacent regions are merged if the distance between them is less than *a merge threshold*. At the end of this step, the regions contain all of the primitives but not all of the pixels; primitives do not cover all of the pixels. Thus, to obtain the final regions, the Voronoi diagram of the primitives, which is the dual of their Delaunay triangulation, is found and the Voronoi polygon of each primitive is included to the region that the primitive belongs to.

5.5 Experiments

We conduct our experiments on 150 images of colon biopsy samples that are randomly taken from the Pathology Department archives of Hacettepe School of Medicine. The samples consist of 5-6 micron thick tissue sections that are stained with hematoxylin-and-eosin, which is routinely used to stain biopsies in hospitals. The images of these samples are taken with a Nikon Coolscope Digital Microscope using $5\times$ microscope objective lens and 1920×2560 image resolution. The tissue images are divided into training and test sets. The training set consists of 50 images that are used to estimate the model parameters. The test set consists of the remaining 100 images that are not used in parameter estimation at all.

Each image is heterogeneous and contains a mixture of normal regions and adenocarcinomatous (cancerous) regions of different grades. The first column of Figs. 5.6–5.9 shows the manual segmentation (gold standard) of these regions provided by a pathologist, who is specialized in colorectal carcinomas. In this figure, normal and adenocarcinomatous regions are labeled as N and AC, respectively[1]. In a tissue image, there may also exist some regions that can be included into

[1]The colon adenocarcinoma originates from epithelial cells and causes organizational changes of these cells, leading to distortions in glands, which are formed of the epithelial cells. To locate adenocarcinomatous regions in a tissue image, regions containing cancerous epithelial cells (and cancerous glands) should be separated from those containing normal epithelial cells (and normal glands). In evaluating segmentation results, it is important how homogeneous segmented regions are in terms of their epithelial cells (and glands).

either a cancerous or a normal region without affecting the medical assessment in the context of colon adenocarcinoma diagnosis. These regions do not contain any epithelial cells (and glands). Thus, they do not affect the assessment in the context of colon adenocarcinoma diagnosis. Such regions are shown with gray shades in Figs. 5.6–5.9.

5.5.1 Evaluation

In our experiments, we provide visual results obtained by the algorithms. Additionally, we quantitatively assess the results using two different criteria: the segmentation accuracy and the number of segmented regions. For computing the accuracy, true positive, false positive, true negative, and false negative pixels are calculated, comparing the segmentation results with the gold standard. Using these pixels, sensitivity and specificity are also computed.

The proposed algorithm and those that we use for comparison are unsupervised, and hence, they do not label their segmented regions. In order to compute the accuracy, we compare each segmented region with the gold standard and label it with the class of the region in the gold standard that mostly overlaps the segmented region (i.e., with the class of the dominant region in the gold standard). Therefore, the pixels located in the non-overlapping parts of the segmented region are considered as either false positive or false negative, depending on the class of its dominant region. Note that, in our evaluations, we do not consider the pixels of regions that could be included in either a normal or a cancerous region.

5.5.2 Comparisons

To investigate the effectiveness of graph run-length matrices (*GraphRLM*), we compare the results of our proposed algorithm with those of four other approaches. In the first approach, we implement the pixel-based counterpart of the proposed algorithm to examine the differences between the use of graph and gray-level run-length matrices. In this approach, the features used in segmentation are

Table 5.2: The parameters of the algorithms and their values that are considered in the estimation of the best parameter sets.

GraphRLM
Window size = $\{32, 64, \mathbf{96}, 128\}$
Distance threshold = $\{0.25, 0.50, ..., \mathbf{1.25}, ..., 3.00, 3.50, 4.00\}$
Component size threshold = $\{10, 25, 50, ..., \mathbf{100}, 150, 200, 250\}$
Merge threshold = $\{\mathbf{0.00}, 0.10, 0.25, 0.50, ..., 3.00, 3.50, 4.00\}$
GrayRLM
Window size = $\{32, 64, 96, 128\}$
Distance threshold = $\{0.25, 0.50, ..., 3.00, 3.50, 4.0\}$
Area threshold = $\{5000, 10000, ..., 60000\}$
Merge threshold = $\{0.00, 0.10, 0.25, 0.50, ..., 3.00, 3.50, 4.00\}$
JSEG
Quantization threshold = $\{10, 25, 50, 75, 100, 150, ..., 450\}$
Scale level = $\{1, 2, 3\}$
Merge level = $\{0.0, 0.1, ..., 0.9\}$
GBS
Gaussian sigma = $\{0.1, 0.2, ..., 1.0\}$
Scale = $\{50, 100, ..., 1000, 1500, ..., 2500\}$
Min area = $\{100, 200, ..., 1000, 1500, ..., 5000, 10000, ..., 25000\}$
ObjSEG
Large window size = $\{96, 128, 160, 192\}$
Small window size = $\{32, 64\}$
Merge threshold = $\{0.00, 1.00, 1.25, ..., 3.50\}$

extracted from gray-level run-length matrices (*GrayRLM*). For that, pixel intensities are quantized into three and seven texture features [49, 96] are defined on the run-length matrices computed at four different angles $(0, \pi/4, 2\pi/4, 3\pi/4)$. The remaining segmentation steps are exactly the same as those of our algorithm except that the grayRLM algorithm uses an area threshold to eliminate small-sized seeds instead of using a component size threshold.

The second approach is our previous work *ObjSEG* defined in Chapter 4, in which we specifically implement the algorithm for the segmentation of histopathological images [45]. This algorithm also employs cytological tissue components to define its texture descriptors; however, it does not use a graph algorithm in its texture definition. Therefore, we include this algorithm in our comparisons to investigate the effectiveness of the use of graphs in texture definition.

The last two approaches are the JSEG algorithm, in which segmentation is achieved by defining a texture descriptor on the quantized pixels [41], and the graph-based algorithm (*GBS*), in which segmentation is achieved by employing a graph constructed over the pixels of an image [44]. We include these algorithms in our comparisons since they have been shown to be effective in many unsupervised segmentation problems although they are not specifically designed for histopathological images.

5.5.3 Parameter selection

All approaches have different model parameters. We estimate the values of these parameters on the training images. To this end, we determine a candidate set for each parameter, try all possible combinations of these candidate sets, and select the one that leads to the best performance on the training images. The parameters of each algorithm and their candidate values are summarized in Table 5.2.

For each algorithm, we first select the best parameter set that leads to the best average accuracy without considering the number of segmented regions. Table 5.3 reports the average segmentation results obtained with such kind of parameter

Table 5.3: The average and standard deviation of segmentation results obtained on the training samples. Parameter sets are selected on the training samples without any restriction on the number of the segmented regions.

	Accuracy	Sensitivity	Specificity	Region no
GraphRLM	99.0 ± 1.3	99.2 ± 1.2	98.7 ± 2.2	76.2 ± 8.8
GrayRLM	98.2 ± 1.9	98.5 ± 2.0	97.4 ± 3.5	50.1 ± 15.8
JSEG	97.0 ± 1.5	95.2 ± 3.7	97.8 ± 1.5	167.7 ± 21.3
GBS	86.0 ± 5.1	83.8 ± 12.0	85.5 ± 12.0	220.6 ± 13.1
ObjSEG	99.0 ± 1.2	98.9 ± 1.7	98.9 ± 2.4	141.1 ± 11.7

selection. Note that all results reported in this subsection are obtained on the training samples. As seen in this table, this selection leads to very high segmentation accuracies, but at the same time, very high number of segmented regions. The main reason of having such high number of regions is that as the accuracy before a merge step cannot be lower than the accuracy after it, the merge parameters are always selected as 0 (i.e., no oversegmented regions are merged). Therefore, we decide to explicitly investigate the effects of a region merge step by calling it with different merge threshold parameters (minimum area parameter of the GBS algorithm, which controls its merge step) right after obtaining the regions. For all of the algorithms, Fig. 5.5 shows the average accuracy and the average number of segmented regions as a function of their merge parameters. This figure shows that a reasonable number of segmented regions can only be obtained with lower accuracy values. When we examine the visual results to understand its reason, we observe that it is not possible to find a common merge parameter that works for all images and it is necessary to select different merge parameters for different images. This is indeed what we observed in our previous work ObjSEG, in which we had to optimize this parameter for each image separately for both the ObjSEG and JSEG algorithms.

Thus, we include the number of segmented regions into the parameter selection criteria, setting an upperbound N on the number and considering only the parameter sets that yield at most N number of segmented regions. As most of the images contain 2-3 regions in the gold standard, we select the value of N as 5 and 10. The upper bound N is used to express the trade-off between the accuracy and the number of segmented regions. Allowing upper bounds that are

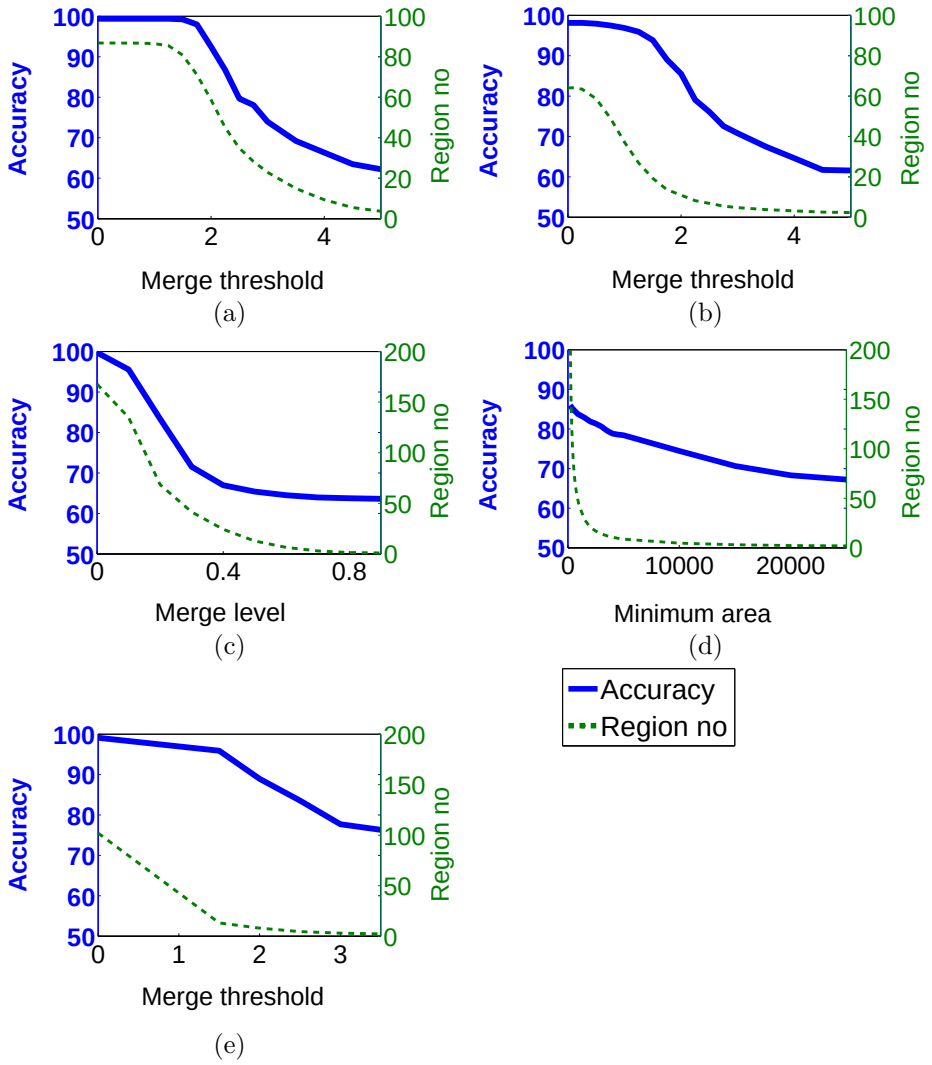


Figure 5.5: The segmentation accuracy and the number of segmented regions as a function of the merge threshold (minimum area) parameter. These results are obtained on the training samples for (a) GraphRLM, (b) GrayRLM, (c) JSEG, (d) GBS, and (e) ObjSEG algorithms.

Table 5.4: The average and standard deviation of segmentation results obtained on the training samples. Parameter sets are selected on the training samples considering only the parameter combinations that give at most (a) 5 regions and (b) 10 regions.

	Accuracy	Sensitivity	Specificity	Region no
GraphRLM	87.1 ± 13.6	90.7 ± 18.7	79.2 ± 33.7	2.9 ± 1.1
GrayRLM	77.2 ± 14.3	74.4 ± 35.8	71.1 ± 38.5	2.9 ± 1.1
JSEG	69.0 ± 12.3	46.8 ± 48.0	72.6 ± 40.2	2.4 ± 1.3
GBS	73.8 ± 8.9	58.6 ± 34.7	77.2 ± 25.1	3.7 ± 0.9
ObjSEG	81.4 ± 14.3	80.0 ± 30.5	76.4 ± 32.8	3.2 ± 0.9

(a)

	Accuracy	Sensitivity	Specificity	Region no
GraphRLM	93.3 ± 7.3	92.8 ± 12.4	91.5 ± 17.9	6.1 ± 1.8
GrayRLM	84.5 ± 14.7	83.7 ± 30.6	78.9 ± 33.7	4.7 ± 2.3
JSEG	87.9 ± 7.9	82.4 ± 22.9	89.7 ± 14.3	6.5 ± 2.1
GBS	77.0 ± 8.4	65.0 ± 26.2	81.5 ± 18.7	6.8 ± 1.7
ObjSEG	87.6 ± 11.8	89.0 ± 19.9	83.8 ± 25.1	5.3 ± 1.6

(b)

greater than the expected number of regions increases the accuracy at the cost of obtaining oversegmented results. Tables 5.4(a) and 5.4(b) report the quantitative results when $N = 5$ and $N = 10$, respectively. These results are different than those reported in Table 4.1 because the merge parameters were selected for each image separately for both the ObjSEG and JSEG algorithms to obtain the results reported in this table. Besides, here more images are used to test the algorithm. Thus, higher accuracies could be obtained in Table 5.4.

Table 5.4(a) shows that the restriction of having at most 5 regions causes lower accuracies. This is attributed to the following two behaviors of the algorithms. They either eliminate some important initial seed regions to start with less number of seeds in the seed determination step. Or they tend to merge heterogeneous segmented regions in the region merge step to keep the number of regions smaller than or equal to 5. Selecting $N = 10$ alleviates the effects of these behaviors, and hence, increases the accuracy of all algorithms. As seen in Table 5.4(b), the number of segmented regions reported for $N = 10$ is much less than the one given in Table 5.3.

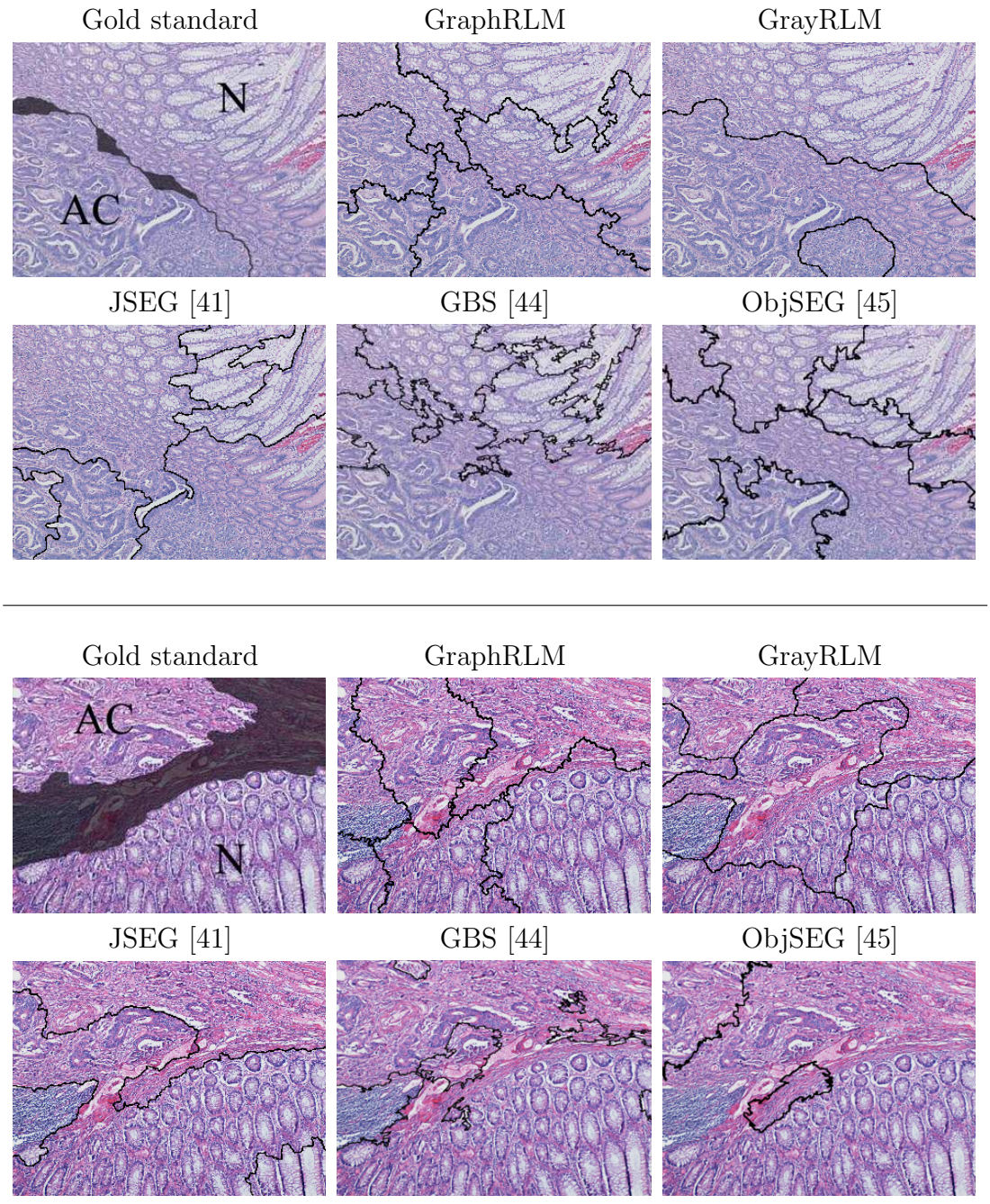


Figure 5.6: The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered.

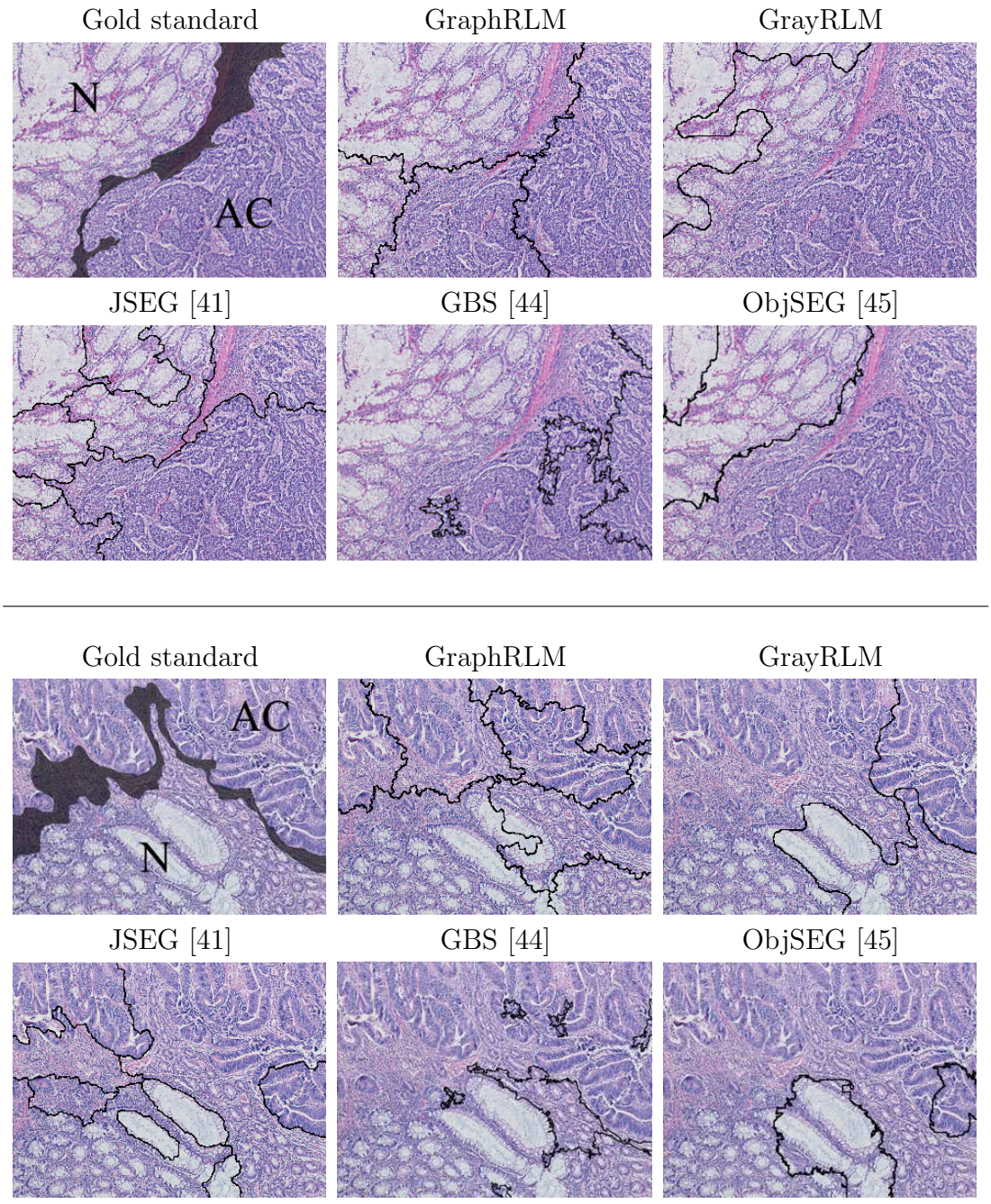


Figure 5.7: The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered.

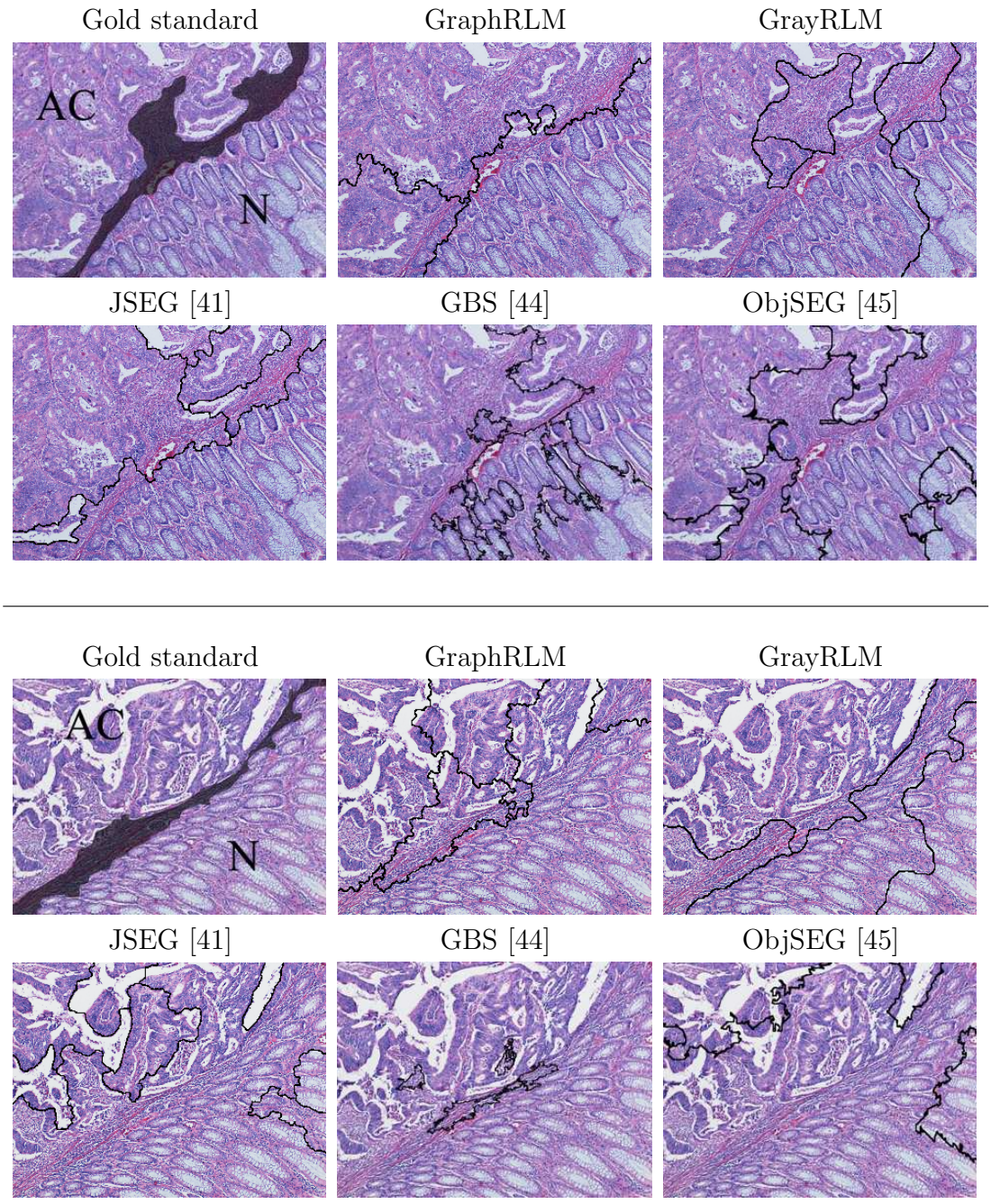


Figure 5.8: The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered.

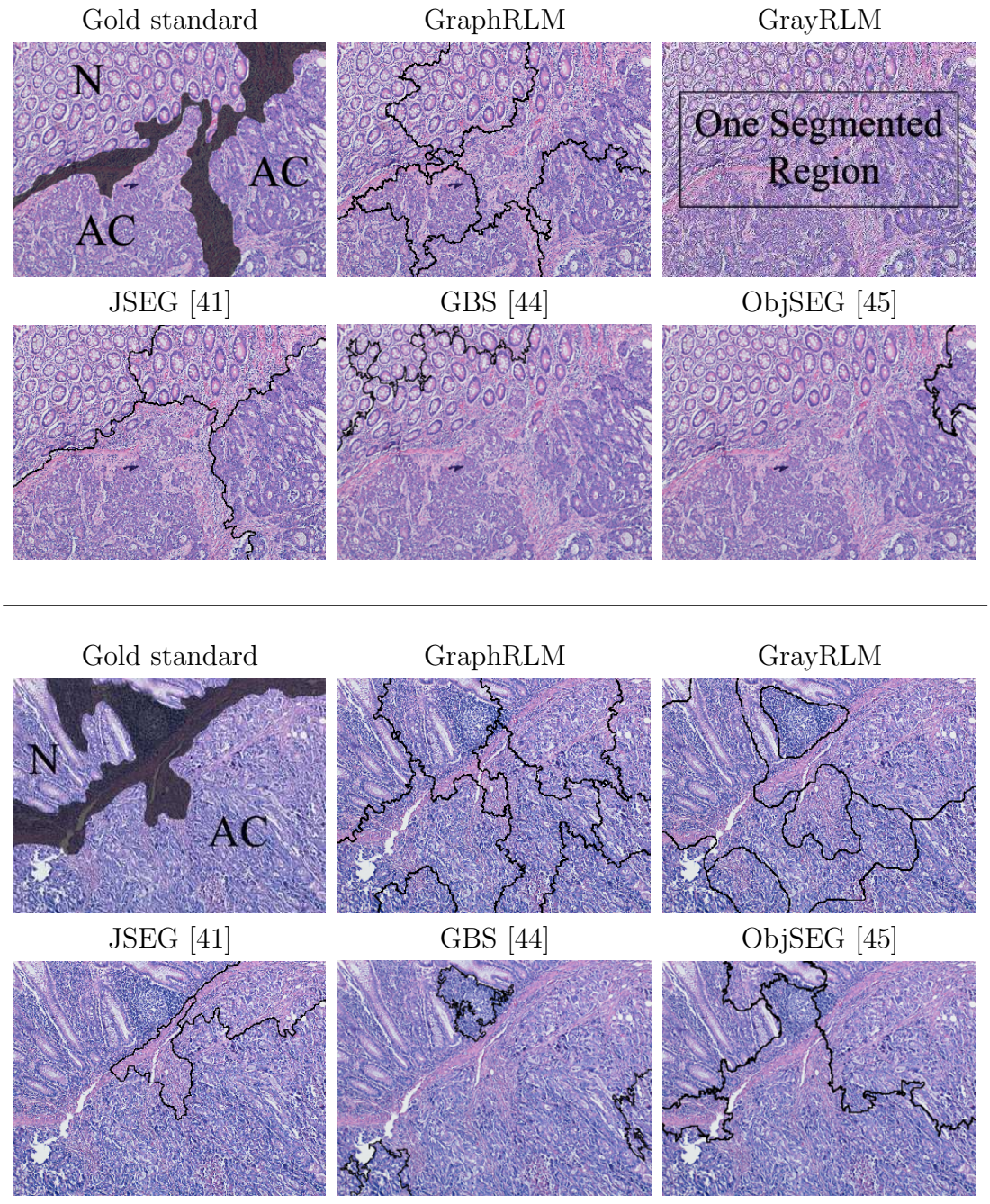


Figure 5.9: The visual results on example images. These results are obtained when only the parameter combinations that give at most 10 regions are considered.

5.5.4 Test results

After selecting the parameters on the training samples, we test our algorithm on the 100 test images. Table 5.5 reports the results. It shows that similar results are obtained for the test samples. Although there is a slight increase in the accuracy of the proposed algorithm for the test samples, the t-test shows that this increase is not statistically significant. This table also shows that our algorithm improves the accuracy of the other algorithms; this improvement is statistically significant with a significance level of 0.05. To understand the reasons of this improvement, we examine the visual results and observe that the other algorithms cause larger variations in their segmentation results; although they are good for some images, they are bad for others. This can be seen on the example images shown in Figs. 5.6–5.9. The variations are indeed due to the difficulty for these algorithms to select a common parameter set that works for all images. On the other hand, the proposed algorithm gives better segmentation results by selecting a better common parameter set that works for more images.

The experiments demonstrate that the proposed algorithm improves the results of the other algorithms. The grayRLM algorithm is implemented as the pixel-based counterpart of our algorithm to investigate the difference between using graph and gray-level run-length matrices. The comparisons show that texture descriptors defined on cytological tissue components using graph run-length matrices is more effective than those defined on pixel intensities using gray-level run-length matrices. The component-based textures are more successful in incorporating the background knowledge into segmentation. The ObjSEG algorithm defines its texture descriptors on cytological tissue components in a different way; it does not employ graphs in this definition. However, the comparisons show that the graphs are more efficient in terms of defining component-based textures. The JSEG and GBS algorithms are the examples of effective algorithms for color-texture image segmentation. The comparisons point to the ill-posedness of the problem. Although algorithms may give good results in general, there is a need of algorithms that are particularly designed for special types of images. Incorporating the background knowledge specific to such images, the segmentation

Table 5.5: The average and standard deviation of segmentation results obtained on the test samples. Parameter sets are selected on the training samples considering only the parameter combinations that give at most 10 regions.

	Accuracy	Sensitivity	Specificity	Region no
GraphRLM	94.8 ± 4.9	95.6 ± 6.5	92.6 ± 13.1	5.9 ± 1.4
GrayRLM	80.9 ± 15.9	82.7 ± 30.9	70.1 ± 39.2	4.3 ± 2.1
JSEG	90.4 ± 6.7	89.3 ± 17.8	88.6 ± 14.9	7.8 ± 2.7
GBS	73.7 ± 9.7	62.3 ± 32.2	76.3 ± 27.5	4.9 ± 1.6
ObjSEG	92.6 ± 9.1	93.1 ± 17.4	89.3 ± 17.3	6.0 ± 1.9

algorithms have the potential of improving their results.

5.5.5 Parameter analysis

The effects of each parameter on the segmentation results are also investigated. For that, three of the four parameters are fixed and the accuracy and the number of segmented regions are observed as a function of the other parameter. In Table 5.2, the selected parameter values are indicated in bold. Fig. 5.10 shows the parameter analysis performed on the test images.

The window size determines the size of a region, on which texture descriptors are defined for a single component. Larger values give too generic descriptors, which make adjacent components more similar. This results in more number of components being grouped in the same seed, and hence, larger but less number of seed regions. In general, less number of initial seeds leads to less number of segmented regions, which usually causes higher segmentation errors. On the other hand, too smaller values give too specific descriptors, which make adjacent components less similar. This gives small-sized initial seeds that are mostly eliminated. This decreases the number of segmented regions, and hence, lowers the accuracy.

The distance threshold determines at what similarity level the components form a single seed. Larger values lead to less number of seeds that are larger in size and contain more dissimilar components. This decreases the accuracy and the number of segmented regions. If it is too small, the components cannot form

large enough seeds that remain uneliminated at the end of the seed determination step. This also decreases the accuracy and the number of segmented regions.

The component size threshold is used to eliminate small-sized components in the seed determination step. If it is too large, most of the components are eliminated. This decreases the number of segmented regions, and hence, the accuracy. If it is too small, small-sized groups are also selected as seeds. This results in very large number of segmented regions at the end, which increases the accuracy.

The merge threshold determines at what similarity level the seeds are merged after the region growing step. Larger values result in more and more seeds being merged into a single region. This decreases the number of segmented regions and the accuracy. On the other hand, smaller values lead to less number of seeds being merged. Hence, the number of segmented regions and the accuracy tend to be higher. In our experiments, this parameter is selected as 0.00, which means that no merge operation is performed. Smaller number of segmented regions (less oversegmented results) could be obtained in two different ways: starting with smaller number of initial seeds at the beginning and/or merging oversegmented regions at the end. In the proposed algorithm, the former one is controlled by the window size, distance threshold, and component size threshold parameters whereas the latter one is controlled by the merge threshold parameter. In our experiments, although the merge threshold is selected to be 0.0, the other parameters are selected such that the algorithm generates at most 10 regions (when $N = 10$). However, the number of segmented regions is comparable with those of the other algorithms, which perform a region merge operation.

In addition to these parameters, the algorithm contains two implicit choices: the grow threshold percentage (grow rate) in region growing and the selection of a dissimilarity measure. In the algorithm, the grow rate is fixed to 0.1. To investigate the effects of its selection, we fix all parameters and change the grow rate from 0.1 to 1.0 in the increments of 0.1. The test results given in Fig. 5.10(e) shows that the grow rate only slightly affects the accuracy and the number of

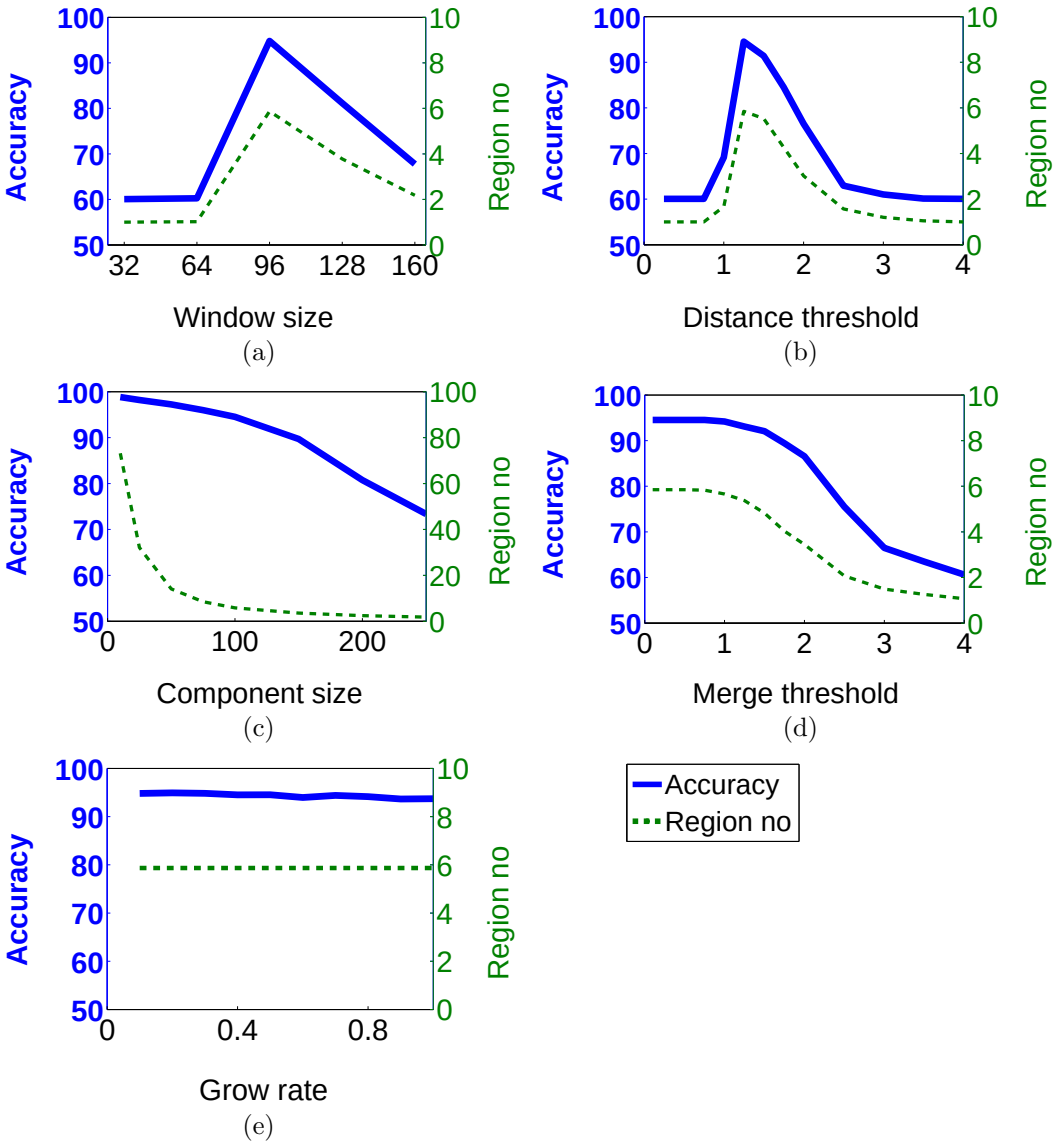


Figure 5.10: The segmentation accuracy and the number of segmented regions as a function of the model parameters: (a) window size, (b) distance threshold, (c) component size threshold, (d) merge threshold, and (e) grow rate percentage.

Table 5.6: Dissimilarity measures analyzed in the experiments.

Euclidean distance	$D(u,v) = \sqrt{\sum_i (u_i - v_i)^2}$
Clark distance	$D(u,v) = \ln \sum_i \sqrt{(u_i \cdot v_i)}$
Bhattacharya distance	$D(u,v) = \sqrt{\sum_i \frac{(u_i - v_i)^2}{(u_i + v_i)^2}}$
K-divergence	$D(u,v) = \sum_i u_i \ln \frac{2\ u_i}{(u_i + v_i)}$

Table 5.7: The effects of a dissimilarity measure on the segmentation accuracy and the number of segmented regions.

	Accuracy	Region no
Euclidean distance	94.8 ± 4.9	5.9 ± 1.4
Clark distance	92.0 ± 9.1	6.2 ± 1.7
Bhattacharya distance	84.5 ± 12.1	4.0 ± 1.4
K-divergence	77.9 ± 13.8	3.9 ± 1.6

segmented regions. On the other hand, the grow rate affects the speed of segmentation. When it is selected as 0.5, the average running time decreases from 52.9 seconds to 30.8 seconds.

To analyze the effects of using different dissimilarity measures, we select four different measures from four different dissimilarity families: (i) Euclidean distance from the Lp Minkowski family, (ii) Bhattacharya distance from the Fidelity family, (iii) K-divergence from the Shannon's entropy family, and (iv) Clark distance from the χ^2 family. The definition of these measures are given in Table 5.6 with $D(u,v)$ being the dissimilarity between feature vectors u and v. In the analysis, we fix all parameters other than the ones that are used to measure dissimilarity and select the others on the training samples. Table 5.7 reports the results obtained on the test samples. It shows that Euclidean and Clark distances give better results. This indicates the importance of using the correct dissimilarity measure. It also shows that one could use different measures to obtain good segmentations.

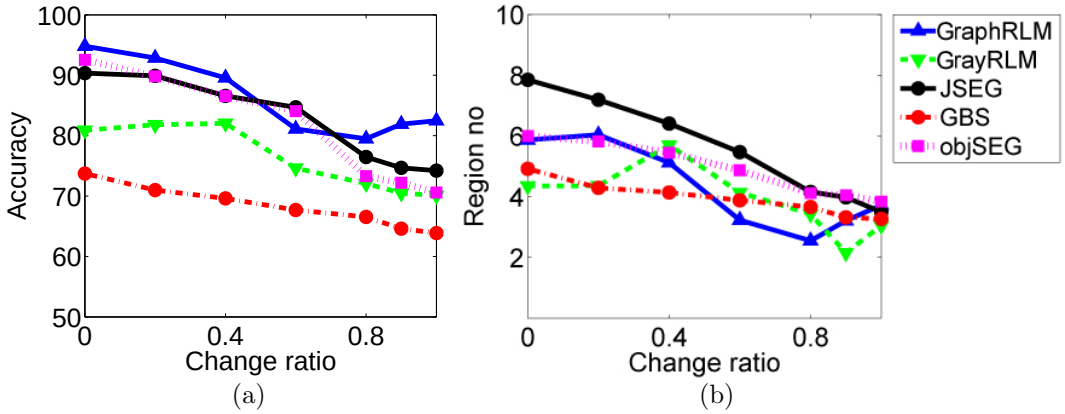


Figure 5.11: The effects of the contrast change ratio on (a) the segmentation accuracy and (b) the number of segmented regions.

5.5.6 Robustness analysis

To understand the robustness of our algorithm with respect to local distortions, we analyze the effects of changes in image contrast on segmentation performance. For that, we distort the test images increasing their contrast. After estimating the parameters on the undistorted training images, we observe the test results as a function of the contrast change ratio (Fig. 5.11). The results of all algorithms show that the accuracy decreases with the increasing contrast change ratios. However, our algorithm still yields high ($\geq$ 90 percent) accuracies when the ratio ≤ 0.4. When the ratio becomes 0.6, it gives lower accuracies especially for some images. We analyze these images and observe that their pink primitives are largely affected by the contrast change such that the number of pink primitives decreases and the remaining ones look like noisy components, which decreases the accuracy. For these images, pink primitives almost disappear when the ratio reaches to 1.0. This disappearance alleviates the look of noisy pink components, and hence, slightly increases the accuracy. Figure 5.11 also shows that our algorithm gives the best accuracies except the case when the ratio is 0.6.

Table 5.8: Computational times of the algorithms.

GraphRLM (grow rate = 0.1)	52.9 ± 3.2
GraphRLM (grow rate = 0.5)	30.8 ± 2.2
GraphRLM (grow rate = 1.0)	26.4 ± 2.8
GrayRLM	172.5 ± 103.4
JSEG	19.1 ± 4.2
GBS	11.7 ± 2.5
ObjSEG (pixel-based growing)	7242.3 ± 265.4
ObjSEG (object based growing)	508.9 ± 107.3

5.5.7 Computational time analysis

The proposed approach first transforms image pixels into a primitive domain and then uses this domain throughout the remaining steps of the segmentation algorithm. Thus, after this transformation, its computational complexity depends on the number of primitives in an image, which is much less than the number of image pixels. The computational time required for processing a single image is 52.9 seconds on the average. This result is obtained on a computer with a Core2Duo 2.8 GHz processor and 3 GB of RAM. As mentioned before, the grow rate affects the speed of segmentation; the computational times for different grow rates are given in Table 5.8. This table also reports the computational times of the other algorithms.

5.6 Summary

This study presents a new algorithm for the unsupervised segmentation of histopathological images. It proposes to incorporate the background knowledge that is specific to histopathological images into segmentation. For this purpose, it introduces a new set of texture descriptors that quantify the spatial distribution of cytological tissue components with the help of a graph constructed on these components.

The proposed algorithm is tested on 150 images of colon tissues that contain normal and cancerous regions. The experiments show that the proposed

algorithm gives accurate segmentation results, providing a reasonable number of segmented regions. Moreover, they also show that the algorithm enables to select a common parameter that leads to good segmentation results. Compared with four other algorithms, the results show that the proposed algorithm is more effective in the segmentation of histopathological images.

Chapter 6

Segmentation of histopathological images using co-occurrence of tissue objects (ObjCooc)

Similarly, this proposed approach relies on characterizing a tissue image with high-level texture features and using them in an efficient segmentation algorithm. It introduces *object co-occurrence features*, which give a new feature set. Although these features are extracted by decomposing a tissue image into a set of objects of different types, as in the case of our previous approaches, it calculates its feature set differently. In this set, features are extracted by calculating the frequency of the co-occurrence of two object types with respect to different distances.

6.1 Feature extraction

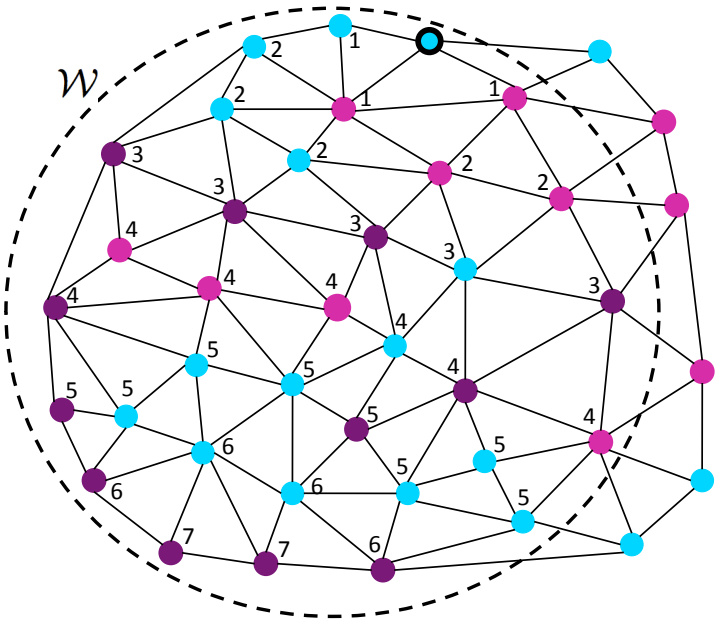
The proposed approach introduces a texture measure to quantify the spatial organization of components in a tissue. Likewise, it transforms a tissue image from the pixel domain to the object domain and defines texture on the tissue components instead of defining it on pixel values. In this new domain, the tissue components are approximately represented by three types of circular objects using the

circle-fitting procedure defined in Chapter 3.

As a result, an image I is decomposed into a set of objects, $O(I) = \{O_i\}$, each of which is represented by its coordinates (x_i, y_i) and its type $t_i \in \{white, pink, purple\}$. Next, each object o_i is characterized by the spatial distribution of objects within its specified neighborhood. For that, a window $\mathcal{W}$ is located at the center of this object and an object co-occurrence matrix C is defined over this window, considering the co-occurrence type between a pair of every object within the window. In this matrix, an entry $C(t_{jk}, d)$ keeps the number of times objects of type t_j co-occur with objects of type t_k at a given distance d. In this work, we construct a Delaunay triangulation on all objects and make use of breadth first traversal on this triangulation to calculate the distance between the objects.

Particularly, to define the object co-occurrence matrix of a window, we consider each object within this window, calculate the distance from this to every other object within the same window using breadth first traversal, and update the matrix accordingly. Fig. 6.1 illustrates the matrix definition for a selected white object that is shown as a thick-bordered cyan circle; here we use cyan circles to represent the white objects. In this figure, the distance from this object to the others, which is computed using breadth first traversal, is indicated next to the corresponding object. The matrix for the selected object is calculated using these distances and the object types. For instance, when $d = 4$, we can reach one white (cyan), four pink, and two purple objects from the selected white object. Thus, $C(\textit{white-white}, 4) = 1$, $C(\textit{white-pink}, 4) = 4$, and $C(\textit{white-purple}, 4) = 2$. To obtain the matrix of the entire window $\mathcal{W}$, this calculation is repeated for every object in $\mathcal{W}$ and their corresponding matrices are accumulated.

After its calculation, we extract 24 features from the object co-occurrence matrix. For that, we define four features from each co-occurrence type by accumulating the co-occurrence values for different distances. For co-occurrence type t_{jk}, these features are defined as follows, with M_d being the maximum distance from one object to another within a window $\mathcal{W}$.



Type \ Distance	1	2	3	4	5	6	7	8	...
White-white	1	3	1	1	6	2	0	0	0
White-pink	2	2	0	4	0	0	0	0	0
White-purple	0	0	4	2	2	2	2	0	0
Pink-pink	0	0	0	0	0	0	0	0	0
Pink-purple	0	0	0	0	0	0	0	0	0
Purple-purple	0	0	0	0	0	0	0	0	0

Figure 6.1: An illustration of object co-occurrence matrix definition for a selected white object shown as a thick-bordered cyan circle. The window $\mathcal{W}$ is located at the center object but the steps illustrated here are repeated for every object in $\mathcal{W}$. Here we select a side object for illustration since we want to show the definition for larger distances (for this side object, the maximum distance is 7, which would be smaller if for example the center object was selected).

$$
\begin{aligned}
\Phi_1 &= C(t_{jk}, 1) \\
\Phi_2 &= C(t_{jk}, 2) + C(t_{jk}, 3) \\
\Phi_3 &= C(t_{jk}, 4) + C(t_{jk}, 5) \\
\Phi_4 &= \sum_{d=6}^{M_d} C(t_{jk}, d)
\end{aligned}
$$

The motivation behind defining the object co-occurrence features is that the relative spatial distribution of the tissue components differs in normal and cancerous regions. For example, in a normal colon tissue, there are single layer epithelial cells around a lumen. In our domain, cell nuclei are represented as purple objects and lumina as white objects. As they are rich in mucin, the cytoplasms of the epithelial cells appear in almost white and are represented as white objects too. Thus, white objects co-occur in a close proximity and surrounded with a single layer of purple objects. In low-grade cancer, single layers of epithelial cells typically turn into multiple layers. They become poor in mucin, which makes their cytoplasms appear in pink. Thus, purple and pink objects are found close to each other. In high-grade cancer, the organization among the tissue components is degenerated. The components, and their corresponding objects, seem to randomly distribute all over a tissue. This makes the co-occurrences of any two object types similar to each other.

The definition of an object co-occurrence matrix is similar to that of a gray-level co-occurrence matrix, which is defined over gray-level pixels to quantify their distribution with respect to a distance and a direction [97]. In contrast with this previous definition, we define our co-occurrence matrices over tissue objects and make use of graph traversals to measure the distance between the objects for quantifying the distribution of tissue components. Since the domain of tissue objects is expected to be less sensitive to small variations and errors at the pixel level, this new definition of the co-occurrence matrix becomes more effective in tissue image representation.

6.2 Segmentation

After transforming an image into the object domain, in which objects $O = \{O_i\}$ are characterized by extracting their object co-occurrence features $\Phi = \{\Phi(O_i)\}$, the segmentation algorithm achieves segmentation on the object domain as well. In this algorithm, we use the idea of obtaining multiple object segmentations and combine them in an ensemble function that is introduced in [98].

This algorithm considers the object segmentation as a graph partitioning problem. For that, it constructs a graph considering the objects as vertices and assigning weighted edges between the vertices by Delaunay triangulation. The weight of an edge corresponds to the similarity of its end points with respect to their features. The weight $w(u, v)$ of an edge $e(u, v)$ is defined as

$$w(u, v) = 1 - \frac{dist(\Phi(u), \Phi(v))}{\max\limits_{ij} dist(\Phi(i), \Phi(j))} \tag{6.1}$$

where $dist(\Phi(u), \Phi(v))$ is the Euclidean distance between the object co-occurrence features of the vertices u and v.

In this work, instead of constructing a graph on the entire set of the objects, a random subset of them is selected according to a *sampling ratio* and the graph is constructed on these selected objects. By doing that, a different graph can be obtained every time the algorithm runs. This helps increase the diversity of segmentations, and thus, the performance of the ensemble.

The proposed multilevel scheme is composed of two main phases: coarsening and uncoarsening. This scheme coarsens the original graph by merging its vertices level-by-level until K' parts (vertices) remain and uncoarsens them by refining the parts at each level. At the end, it outputs a partition vector containing the part each vertex belongs to.

At the end of the uncoarsening phase, a partition only on the selected vertices (objects) is obtained since the multilevel graph partitioning algorithm takes a random subset of the objects, but not all of them, every time it runs. Then, this partition is used to induce a complete partition on the entire object set

as follows: First the selected neighbor objects of each unselected object O_i is determined using a Delaunay triangulation and then O_i are assigned to the part that contains its most similar neighbor object.

This segmentation procedure is processed $\mathcal{M}$ times in order to obtain different segmentations. Then, multiple segmentation results of the multilevel graph partitioning algorithm are combined by utilizing a cluster ensemble framework [98]. At the end of the ensembling procedure, since the segments are obtained on objects but not on image pixels, the pixels are mapped to the segmented objects by assigning each pixel to its closest object, and obtain the segmented image regions.

6.3 Experiments

In our experiments, we use 200 microscopic images of colon biopsy samples stained with hematoxylin-and-eosin. These biopsies are randomly selected from the Pathology Department Archives of Hacettepe Medical School. The images are taken using a Nikon Coolscope Digital Microscope; the microscope objective lens is $5\times$ and the image resolution is 1920×2560. The images are so selected as to contain both normal regions and adenocarcinomatous (cancerous) regions of different grades. The selected images are randomly divided into training and test sets. The training set includes 50 tissue images and is used to estimate the parameters of the algorithms. The test set includes the remaining 150 tissue images. In this work, we increase the number of test images from 100 to 150. Therefore, the results of the algorithms that we use in our comparisons are different than those reported in Chapter 5.

6.3.1 Evaluation

We quantitatively evaluate all the algorithms by comparing their results with the gold standard provided by our medical collaborator. As the algorithms are unsupervised, we obtain the quantitative results as follows: First, we assign each segmented region S to the label of its most overlapping region R in the gold standard. Then, we consider the overlapping pixels of S and R as either true positive (TP) or true negative (TN) and the non-overlapping pixels of S as either false positive (FP) or false negative (FN), according to the label of R (e.g., if R is cancerous, the overlapping pixels are considered as TP and the non-overlapping pixels as FP). Subsequently, we compute the *F-score* as the evaluation criterion for comparison and parameter selection. We select the parameter set that maximizes the average *F-score* on training images. For our algorithm, the parameter sets considered in parameter selection are all combinations of the following values: size of the window $\mathcal{W} = \{32, 64, 96, 128\}$, coarsest graph size $K' = \{2, 3, 4, 5, 10, ..., 50\}$, *sampling ratio* $= \{0.01, 0.02, 0.05, 0.10, ..., 0.30, 0.50, 0.70, 0.90, 1.00\}$, and number of segmentations $\mathcal{M} = \{50, 100, ..., 300, 400, 500\}$. The parameter sets of the other algorithms can be found in Table 5.2. Note that we select the parameters separately for each number K of the segmented regions.

We obtain the gold standards considering colon adenocarcinoma, which accounts for 90-95 percent of all colorectal cancers. As this cancer type originates from epithelial cells, which form colon glands, it causes deformations in the gland structures. Thus, as mentioned before, to obtain its gold standard, an image is segmented into regions containing normal and cancerous glands (epithelial cells). In addition to these, there may exist inflammatory and stromal regions that do not contain any glands (as shown with green shades in Figs. 6.2–6.5). In our experiments, we do not consider such regions in F-score calculations.

6.3.2 Results

The multilevel segmentation (*ObjCooc*) algorithm segments an image into K regions. This is opposed to the case, in which the number of segmented regions is

Table 6.1: The average test set results obtained by the proposed *ObjCooc* algorithm and their standard deviations.

K	Accuracy	Sensitivity	Specificity	F-score
2	92.0 ± 9.0	92.6 ± 15.9	87.8 ± 23.8	91.7 ± 14.0
3	92.9 ± 7.6	94.2 ± 12.4	90.3 ± 15.8	92.9 ± 10.7
4	94.2 ± 5.5	94.6 ± 8.6	92.9 ± 9.0	94.4 ± 6.0
5	94.9 ± 5.5	95.8 ± 7.1	92.9 ± 9.3	95.2 ± 5.6

dynamically selected by an algorithm; the ones that we use in our comparisons are the examples of such algorithms. The dynamic selection of this number usually depends on an external parameter (e.g., a merge threshold). However, it is hard to select a common value of this parameter such that it gives good results for multiple images in terms of both the accuracy and the number of segmented regions. The values that give accurate results for multiple images typically lead to oversegmentations. Thus, in order to prevent such oversegmentations, we externally set the number K of regions in our algorithm.

In the experiments, we select K as 2 and 3 since the tissue images we use have 2 or 3 regions to be segmented. Table 6.1 reports the average test results for different K values. It shows that the *ObjCooc* algorithm can achieve accurate results, $> 90\%$ accuracy and $> 90\%$ F-score on the average, even when $K = 2$ and 3. Larger values of K slightly improve the results.

6.3.3 Comparisons

We compare the proposed *ObjCooc* algorithm with two sets of algorithms. The first set includes our earlier studies, the *GraphRLM* [46] defined in Chapter 5 and *ObjSEG* [45] defined in Chapter 4, that are also implemented for histopathological image segmentation. As explained in the previous chapter, these studies quantify an image defining different sets of features on the tissue objects and achieve segmentation using a region growing algorithm. We make these comparisons to understand the effectiveness of the newly introduced object co-occurrence features as well as our multilevel segmentation algorithm.

Table 6.2: The average test set results obtained by the other algorithms and their standard deviations. The parameters of the algorithms are selected considering an upperbound N on the number of segmented regions.

		Accuracy	Sensitivity	Specificity	F-score	Region no
$N \leq 5$	GraphRLM	84.8 ± 14.4	85.8 ± 26.2	76.2 ± 35.7	81.5 ± 26.6	2.8 ± 1.1
	ObjSEG	86.9 ± 11.5	90.4 ± 21.8	77.2 ± 28.5	78.6 ± 27.9	4.1 ± 1.3
	GBS	73.4 ± 8.9	64.5 ± 33.8	72.2 ± 30.3	58.5 ± 37.7	3.7 ± 1.2
	JSEG	69.4 ± 12.1	62.7 ± 45.5	62.2 ± 39.8	46.2 ± 38.4	2.9 ± 1.3
$N \leq 10$	GraphRLM	91.5 ± 9.8	92.3 ± 15.6	87.3 ± 24.2	91.3 ± 13.8	5.6 ± 1.6
	ObjSEG	89.9 ± 11.4	89.3 ± 23.5	85.8 ± 22.7	87.8 ± 19.1	5.8 ± 2.0
	GBS	74.0 ± 9.5	64.7 ± 31.9	74.5 ± 28.7	61.9 ± 32.0	5.0 ± 1.7
	JSEG	89.6 ± 7.2	89.1 ± 16.3	87.1 ± 18.2	87.6 ± 12.5	7.9 ± 2.7

The second set includes two algorithms: the graph-based segmentation (*GBS*) [44] and *JSEG* [41] algorithms that are not specifically implemented for histopathological images but are known as effective segmentation algorithms for images in general. We make these comparisons to understand the importance of using domain specific knowledge in segmentation.

None of these algorithms take the number of segmented regions externally; instead, they select this number dynamically for each image according to their parameters. Each algorithm selects the parameter set that gives the highest F-score on the training images. This selection maximizes this measure at the expense of obtaining more segmented regions. Thus, to prevent oversegmentations, we enforce the algorithms not to consider the parameter sets that give more regions than an upperbound N. In our previous study, *GraphRLM*, we selected $N = 5$ and 10. Table 6.2 provides the test results for these upperbounds. The results show that the selection of N greatly affects the segmentation quality. When it is set to a smaller value to obtain less oversegmented results, the performance significantly drops. When it is set to a larger value, such as $N = 10$, the performance increases but the results tend to be oversegmented. Comparing the results of the *ObjCooc* algorithm with those given in this table, we observe that the *ObjCooc* algorithm achieves higher quality even when $K = 2$ or 3.

To make a fairer comparison, we also modify the algorithms so that they segment an image into exactly K regions. For that, we enforce the *GraphRLM*

Table 6.3: The average test set results obtained by the *ObjCooc*, *GraphRLM*, *ObjSEG*, and *GBS* algorithms and their standard deviations.

		Accuracy	Sensitivity	Specificity	F-score
$K = 2$	ObjCooc	92.0 ± 9.0	92.6 ± 15.9	87.8 ± 23.8	91.7 ± 14.0
	GraphRLM	83.5 ± 13.1	81.1 ± 27.1	79.1 ± 32.6	80.8 ± 22.8
	ObjSEG	82.9 ± 11.7	86.9 ± 22.9	69.8 ± 36.7	82.7 ± 18.9
	GBS	75.5 ± 11.0	71.5 ± 36.4	71.2 ± 28.8	56.0 ± 32.7
$K = 3$	ObjCooc	92.9 ± 7.6	94.2 ± 12.4	90.3 ± 15.8	92.9 ± 10.7
	GraphRLM	87.6 ± 10.6	90.4 ± 15.7	80.1 ± 26.9	88.1 ± 12.9
	ObjSEG	88.4 ± 8.3	90.6 ± 15.9	81.5 ± 23.7	88.5 ± 11.9
	GBS	76.3 ± 10.2	70.2 ± 32.9	74.5 ± 28.4	58.0 ± 33.3

algorithm to select the largest K initial seeds in its seed determination step. We enforce the other algorithms to dynamically select their merge parameters for each image so that their merging step merges the regions to each other until K regions are left. Table 6.3 reports the test results when $K = 2$ and 3. Here we do not report the results of the *JSEG* algorithm since it gives very inaccurate results; some of its results are depicted in Figs. 6.2–6.5. As possible future work, one could consider to modify the other steps of the *JSEG* algorithm to obtain exactly K segmented regions with better quality. Table 6.3 shows that the *ObjCooc* algorithm greatly improves the results of the others. The t-test indicates that this improvement is statistically significant with $\alpha = 0.05$. Figs. 6.2–6.5 give visual results on some example images.

6.3.4 Discussion

The *ObjCooc* algorithm segments an image into a selected number K of regions. Of course, the selection of K is closely related to the application of interest. For applications in which such a value cannot be defined, dynamic selection should be incorporated. In the experiments, we also implement an extended version of our algorithm that selects K dynamically. To this end, starting with $K = 2$, we compute an invariant criterion for the segmentation result and find the first K for which the criterion falls below a certain threshold. We use the invariant criterion $J = |S_W|/|S_B|$, where S_W and S_B correspond to the within-segmentation (cluster) and between-segmentation scatter matrices [99]. Selecting a threshold value as

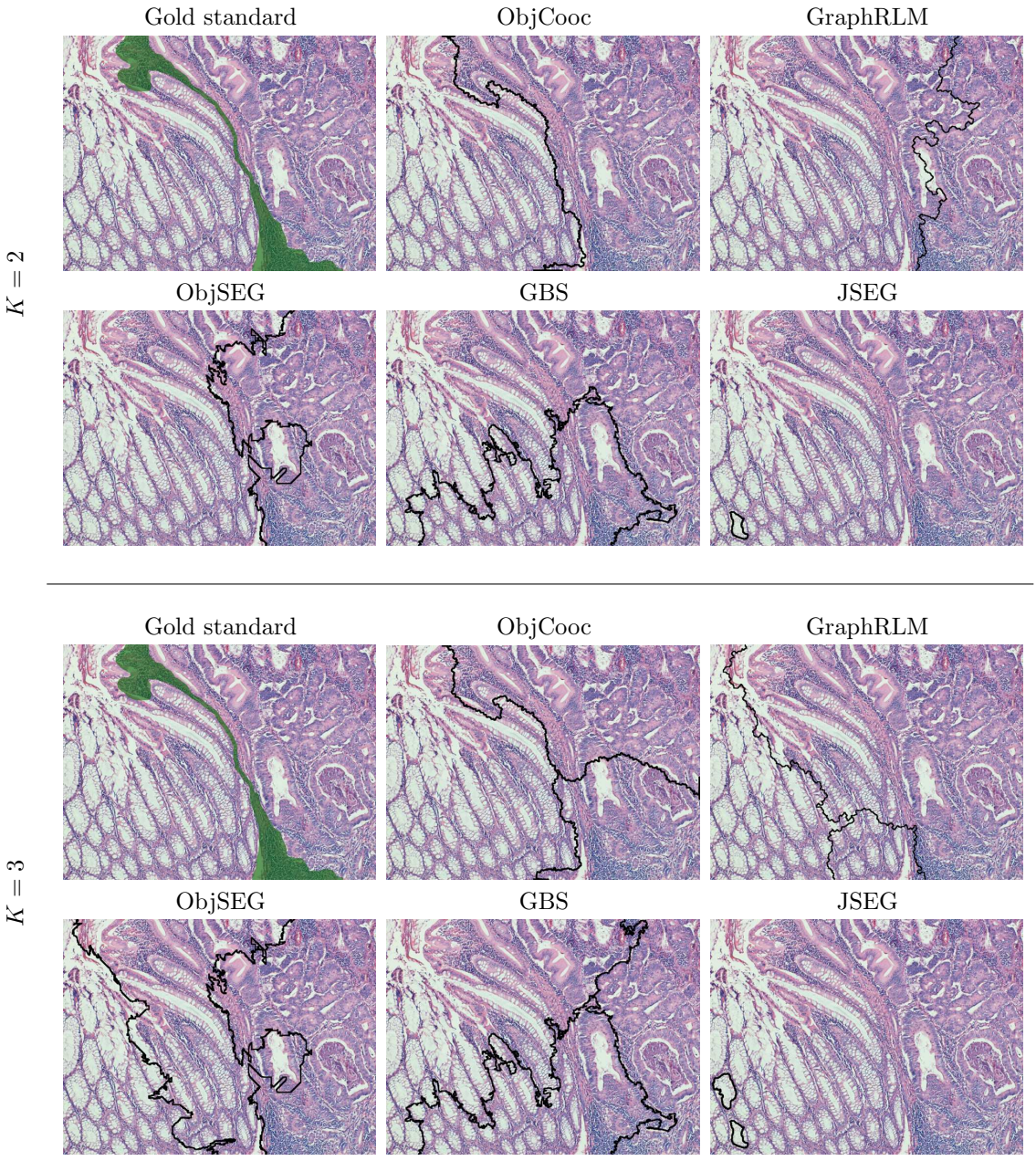


Figure 6.2: Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3.

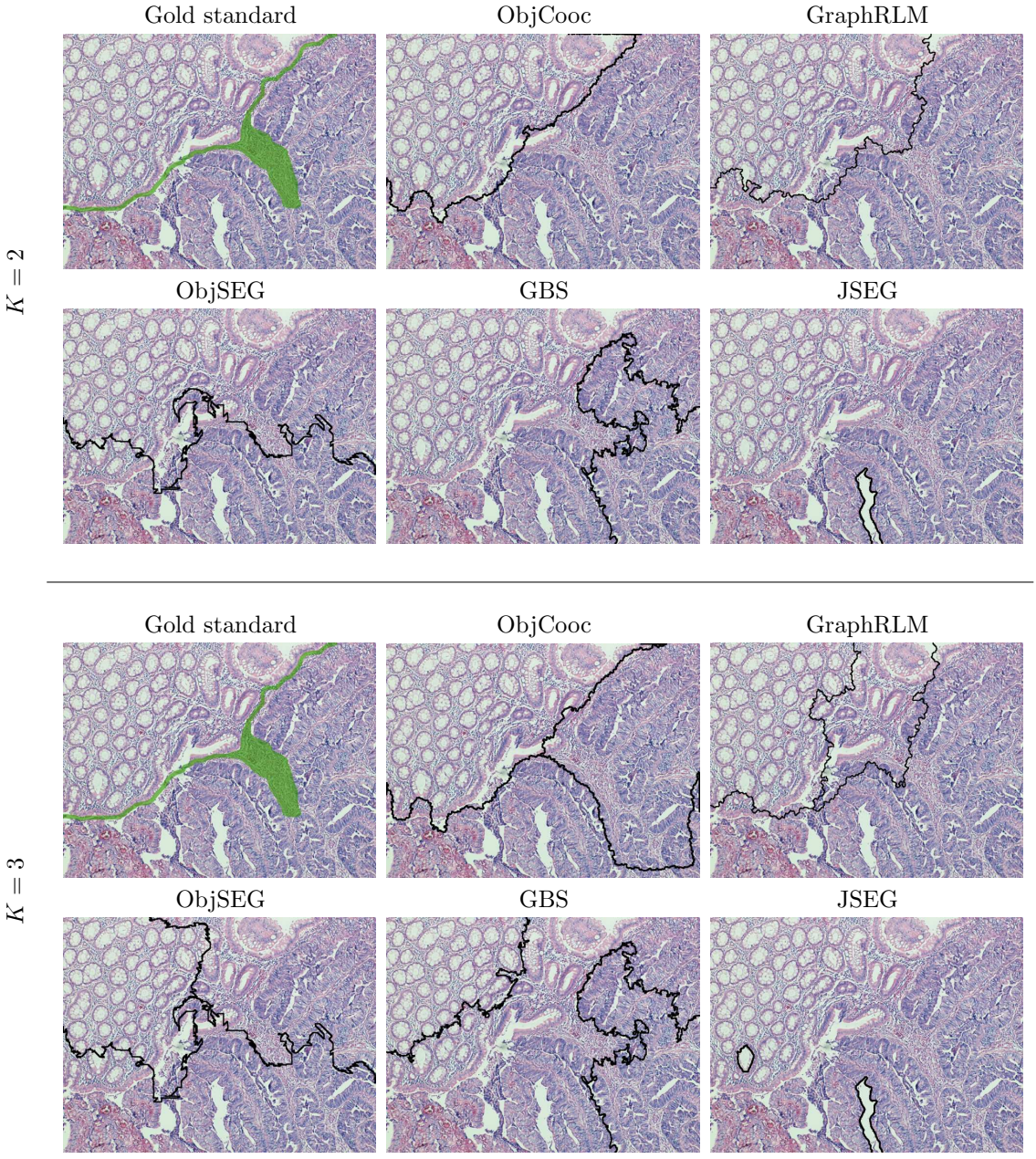


Figure 6.3: Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3.

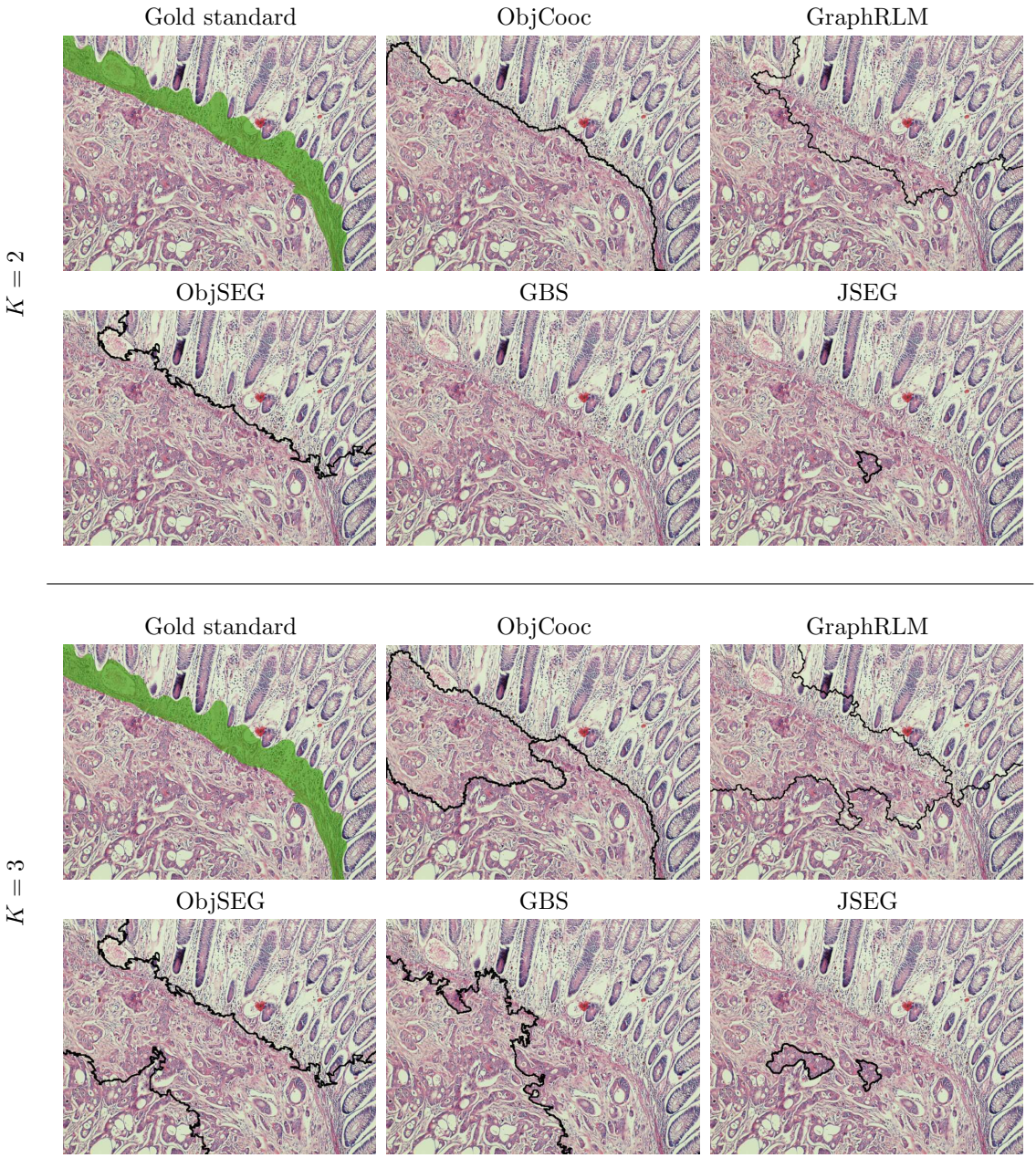


Figure 6.4: Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3.

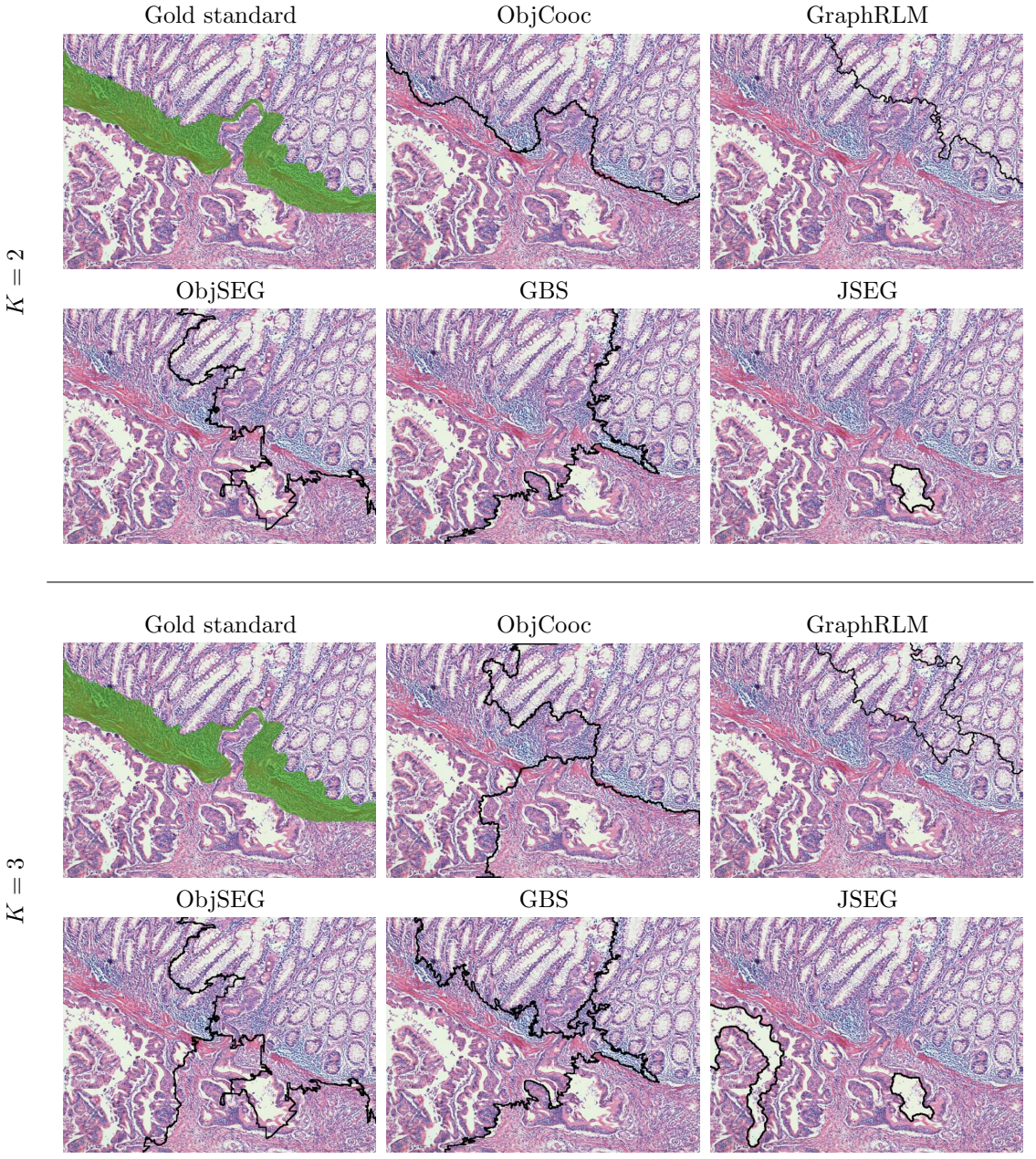


Figure 6.5: Visual results obtained on example tissue images when the number K of segmented regions is set to 2 and 3.

Table 6.4: The average test set results obtained by the *ObjCooc* algorithm that uses an alternative co-occurrence matrix calculation.

K	Accuracy	Sensitivity	Specificity	F-score
2	88.9 ± 11.0	89.6 ± 18.0	82.4 ± 30.4	88.4 ± 17.2
3	91.6 ± 7.7	91.8 ± 13.8	87.7 ± 20.5	91.3 ± 12.0
4	92.9 ± 6.4	94.2 ± 7.3	90.1 ± 13.2	93.4 ± 6.7
5	93.9 ± 5.4	95.3 ± 7.0	90.7 ± 9.8	94.3 ± 5.8

0.2, we obtain 94.1 ± 6.4 percent accuracy and 94.1 ± 9.8 percent F-score, on the average. The average number of regions is 3.3 ± 1.1. These results indicate the potential use of the algorithm for variable K values.

The *ObjCooc* algorithm makes use of Delaunay triangulation to calculate a co-occurrence matrix. It is also possible to use different approaches for this calculation. For instance, for an object, one may compute the Euclidean distance from this object to every other object in a window $\mathcal{W}$ and create a histogram by binning. The co-occurrence matrix is then calculated by accumulating the histograms of the objects that fall in $\mathcal{W}$ based on their object types and the same object co-occurrence features are extracted from this matrix. Table 6.4 reports the test set results obtained by this approach. These results are slightly less than those obtained by the proposed co-occurrence matrix calculation. The results indicate the effectiveness of the use of Delaunay triangulation in extracting descriptive features.

Although the experiments are conducted on the images of hematoxylin-and-eosin stained tissues to locate normal and colon cancerous regions, the proposed algorithm has a potential to be used on different types of histology images as well as to locate regions of different characteristics. Fig. 6.6 shows the results when the algorithm is applied to two example images of tissues stained with immunohistochemistry. In these preliminary results, we observe that normal and cancerous regions are successfully segmented.

Moreover, we run our algorithm to locate different types of regions. As an example, the first column of Fig. **??** shows an image with four different regions: normal region (marked as 1), inflammatory region (marked as 2), and cancerous regions of different grades (marked as 3 and 4). The *ObjCooc* algorithm is

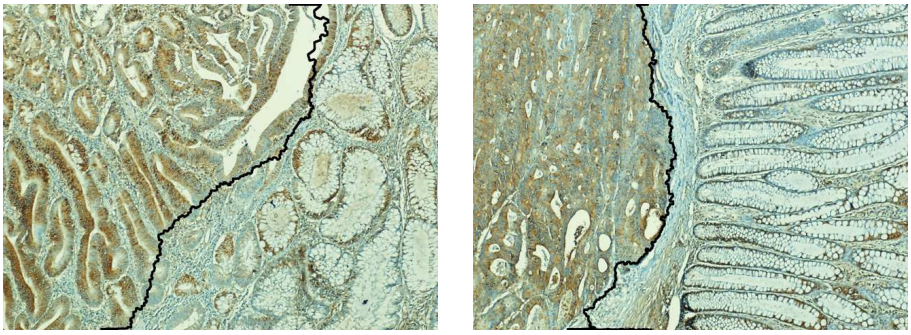


Figure 6.6: The visual results obtained on the example images of tissues stained with immunohistochemistry.

successful to segment these regions. As another example, the second column of Fig. **??** shows an image that contains a region of dysplastic glands (marked as 1 and 2). For this image, we observe that such regions can only be roughly segmented. To segment dysplastic glands accurately, one may consider to combine pixel-level textures to the proposed features. This would be an interesting future research direction.

For the case of computational performance of the algorithm, the computational time for a single image takes approximately 7-8 minutes using a computer with a Intel Xeon 2.27 GHz processor and 12 GB of RAM. Since our implementation uses Java for feature extraction and Matlab for multilevel segmentation, it is possible to obtain significant speedups by implementing the algorithm with C or C++ and using an optimized compiler.

6.4 Summary

This study presents a new methodology for unsupervised segmentation of histopathological images. This method defines a set of new high-level texture descriptors, called *object co-occurrence features*, to represent the prior knowledge in a tissue. The object co-occurrence features are expected to be less vulnerable to noise and variations that are typically observed at the pixel-level of tissue images. This feature set is used in a multilevel segmentation algorithm, in which

image segmentation is achieved by partitioning the objects according to their co-occurrence features.

In our experiments we tested the proposed approach on 200 colon tissue images. The experimental results showed that the proposed algorithm is effective to obtain higher quality results compared to the other algorithms even the segmented number of regions is limited to 2 and 3.

Chapter 7

Conclusion

This thesis introduced three new texture descriptors, namely ObjSEG [45], GraphRLM [46], and ObjCooc [47] textures, for the first time. These texture descriptors are extracted on tissue components that are modeled by circular objects. These objects approximately represent the components, including epithelial tissue components (nuclei), connective tissue components (stroma), and luminal structures (lumen). Since these object-oriented texture descriptors are defined on the tissue components, they represent the spatial organization of the components better than their previous counterparts, which use pixel-level descriptors for quantizing a tissue. Following the extraction of object-oriented texture descriptors, image segmentation algorithms that use these descriptors for segmenting histopathological tissue images are implemented.

In the work of ObjSEG, we proposed a new homogeneity measure based on the distribution of the objects. Using this object-oriented measure, we demonstrated a new object-based segmentation algorithm. As opposed to the existing algorithms that use pixel-based information in defining their homogeneity measure, our segmentation algorithm uses object-based information, for the very first time.

Working with colon biopsy images with similar color distributions in their

heterogeneous regions, we demonstrated that our object-oriented algorithm significantly improves the accuracy in segmenting tumorous regions and also other non-cancerous tissue transformations compared to its pixel-based counterpart.

Later, we improved the performance of ObjSEG features by proposing a new region-growing algorithm for the unsupervised segmentation of histopathological images. This algorithm introduced a new region growing approach, in which the growing process relies on the relationship information of objects that represent cytological components within a tissue. Our experimental results showed that the proposed region growing algorithm leads to better segmentation results compared to ObjSEG.

Since we obtain better results with the use of relationships of objects, we presented a new algorithm, called GraphRLM, for unsupervised segmentation of histopathological images. It introduced a new set of texture descriptors based on run-length matrices that quantify the spatial distribution of cytological tissue components with the help of a graph constructed on these components. This method is the first study in literature that defines run-length matrices over graphs. GraphRLM is tested on 150 images of colon tissues that contain normal and cancerous regions. The experiments showed that GraphRLM gives accurate segmentation results, providing a reasonable number of segmented regions. Moreover, they also showed that the algorithm enables to select a common parameter set that leads to good segmentation results.

To improve the segmentation results using relationship information of objects we proposed ObjCOOC, where we defined a set of new high-level texture descriptors based on co-occurrence of objects to represent the prior knowledge in a tissue and uses them in an effective multilevel segmentation algorithm. The experiments on 200 colon tissue images show that the proposed algorithm is effective to obtain higher quality results compared to the other algorithms.

For the future perspective of this thesis, the proposed segmentation algorithms can be used as a part of a two-phase algorithm for whole slide image segmentation. In the first phase, empty areas of a whole slide can be separated from histological sections using a simple algorithm. Then, in the second phase, the

histological sections can be segmented into their homogeneous regions using the proposed segmentation algorithms. Since a typical slide contains 75-100 5× images of histological sections and a single segmentation process may take up to 10 minutes. One of the future work might be to speed up the algorithm by using parallel computing solutions in several steps of the algorithms; such as object decomposition, feature extraction, and region growing steps.

Another future research direction is to use the proposed texture descriptors for supervised classification. The descriptors extracted from the segmented regions could be used for cancer diagnosis and grading. Although the algorithm is particularly designed for histopathological images, the new texture definitions proposed in this thesis have a potential to be used in different applications such as remote sensing image analysis. For that, one could define primitives on the dominant colors in a similar way and construct a graph to define the texture of these primitives. This would be another future research direction of the thesis.

On the other hand, one can investigate the effect of using different shapes of primitive objects, such as ellipses, to represent the tissue components better. However, using different shapes may have a negative effect on the computational time of extracting the objects. One can also study the effect of using information of *groups of objects* that have special properties; such as forming a complete graph or cliques of the same type of objects, forming a circular distribution around one object with a particular angle distribution, and forming a dense/sparse neighborhood of objects in a particular area.

Bibliography

[1] I. D. Nagtegaal and J. H. J. M. van Krieken, "The role of pathologists in the quality control of diagnosis and treatment of rectal canceran overview," *European Journal of Cancer*, vol. 38, no. 7, pp. 964 – 972, 2002.

[2] P. W. Huang and C. H. Lee, "Automatic classification for pathological prostate images based on fractal analysis," *IEEE Transactions on Medical Imaging*, vol. 28, no. 7, pp. 1037 – 1050, 2009.

[3] D. Altunbay, C. Cigir, C. Sokmensuer, and C. Gunduz-Demir, "Color graphs for automated cancer diagnosis and grading," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 3, pp. 665 – 674, 2010.

[4] S. Doyle, M. D. Feldman, J. Tomaszewski, and A. Madabhushi, "A boosted bayesian multiresolution classifier for prostate cancer detection from digitized needle biopsies," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1205 – 1218, 2012.

[5] L. Zheng, A. W. Wetzel, J. Gilbertson, and M. J. Becich, "Design and analysis of a content-based pathology image retrieval system," *IEEE Transactions on Information Technology in Biomedicine*, vol. 7, no. 4, pp. 249 – 255, 2003.

[6] F. Yu and H. H. Ip, "Semantic content analysis and annotation of histological images," *Computers in Biology and Medicine*, vol. 38, no. 6, pp. 635 – 649, 2008.

[7] Y. Al-Kofahi, W. Lassoued, W. Lee, and B. Roysam, "Improved automatic detection and segmentation of cell nuclei in histopathology images," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 4, pp. 841 – 852, 2010.

[8] C. Gunduz-Demir, M. Kandemir, A. B. Tosun, and C. Sokmensuer, "Automatic segmentation of colon glands using object-graphs," *Medical Image Analysis*, vol. 14, no. 1, pp. 1 – 12, 2010.

[9] J. Smolle, "Computer recognition of skin structures using discriminant and cluster analysis," *Skin Research and Technology*, vol. 6, no. 2, pp. 58 – 63, 2000.

[10] M. Mete, X. Xu, C. Y. Fan, and G. Shafirstein, "Automatic delineation of malignancy in histopathological head and neck slides.," *BMC Bioinformatics*, vol. 8, no. S-7, 2007.

[11] Y. Wang, D. Crookes, O. S. Eldin, S. Wang, P. Hamilton, and J. Diamond, "Assisted diagnosis of cervical intraepithelial neoplasia (cin)," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 1, pp. 112 – 121, 2009.

[12] D. Romo, E. Romero, and F. Gonzalez, "Learning regions of interest from low level maps in virtual microscopy," *Diagnostic Pathology*, vol. 6, no. Suppl 1, p. S22, 2011.

[13] A. Andrion, C. Magnani, P. G. Betta, A. Donna, F. Mollo, M. Scelsi, P. Bernardi, M. Botta, and B. Terracini, "Malignant mesothelioma of the pleura: inter-observer variability," *Journal of Clinical Pathology*, vol. 48, pp. 856 – 860, 1995.

[14] N. Krieger, R. A. Hiatt, R. W. Sagebiel, W. H. Clark Jr., and M. C. Mihm Jr., "Inter-observer variability among pathologists' evaluation of malignant melanoma: Effects upon an analytic study," *Journal of Clinical Epidemiology*, vol. 47, no. 8, pp. 897 – 902, 1994.

[15] H. Struikmans, C. Wrlm-Rodenhuis, T. Stam, G. Stapper, R. J. H. A. Tersteeg, G. H. Bol, and C. P. J. Raaijmakers, "Interobserver variability of clinical target volume delineation of glandular breast tissue and of boost volume in tangential breast irradiation," *Radiotherapy and Oncology*, vol. 76, no. 3, pp. 293 – 299, 2005.

[16] M. Fukunaga, H. Katabuchi, T. Nagasaka, Y. Mikami, S. Minamiguchi, and J. M. Lage, "Interobserver and intraobserver variability in the diagnosis of hydatidiform mole," *The American Journal of Surgical Pathology*, vol. 29, no. 7, pp. 942 – 947, 2005.

[17] L. L. Gunderson, J. M. Jessup, D. J. Sargent, F. L. Greene, and A. K. Stewart, "Revised tn categorization for colon cancer based on national survival outcomes data.," *Journal of Clinical Oncology*, 2009.

[18] National Institute for Health and Clinical Excellence, "Referral guidelines for suspected cancer–nice guideline," 2005.

[19] J. Barret, M. Jiwa, P. Rose, and W. Hamilton, "Pathways to the diagnosis of colorectal cancer: an observational study in three uk cities," *BMC Family Practice*, vol. 23, no. 1, pp. 15 – 19, 2006.

[20] L. H. Sobin, C. Wittekind, and International Union against Cancer, *TNM classification of malignant tumours*. UICC Series, Wiley-Liss, 1997.

[21] C. Wittekind and L. Sobin, *TNM classification of malignant tumours*. Wiley-Liss, 2002.

[22] M. Wiltgen, A. Gerger, and J. Smolle, "Tissue counter analysis of benign common nevi and malignant melanoma," *International Journal of Medical Informatics*, vol. 69, no. 1, pp. 17 – 28, 2003.

[23] A. Nasser Esgiar, R. N. G. Naguib, B. S. Sharif, M. K. Bennett, and A. Murray, "Microscopic image analysis for quantitative measurement and feature identification of normal and cancerous colonic mucosa," *IEEE Transactions on Information Technology in Biomedicine*, vol. 2, no. 3, pp. 197 – 203, 1998.

[24] B. Weyn, G. V. de Wouwer, M. Koprowski, A. V. Daele, K. Dhaene, P. Scheunders, W. Jacob, and E. V. Marck, "Value of morphometry, texture analysis, densitometry and histometry in the differential diagnosis and prognosis of malignant mesothelioma," *Journal of Pathology*, vol. 4, no. 189, pp. 581 – 589, 1999.

[25] C. Demir, S. H. Gultekin, and B. Yener, "Learning the topological properties of brain tumors," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 2, no. 3, pp. 262 – 270, 2005.

[26] O. Sertel, J. Kong, U. Catalyurek, G. Lozanski, J. Saltz, and M. Gurcan, "Histopathological image analysis using model-based intermediate representations and color texture: Follicular lymphoma grading," *Journal of Signal Processing Systems*, vol. 55, pp. 169 – 183, 2009. 10.1007/s11265-008-0201-y.

[27] E. Ozdemir, C. Sokmensuer, and C. Gunduz-Demir, "A resampling-based markovian model for automated colon cancer diagnosis," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 1, pp. 281 – 289, 2012.

[28] G. T. Deans, P. W. Hamilton, P. C. H. Watt, M. Heatley, K. Williamson, C. C. Patterson, B. J. Rowlands, G. Parks, and R. Spence, "Morphometric analysis of colorectal cancer," *Diseases of the Colon and Rectum*, vol. 36, pp. 450 – 456, 1993.

[29] J. K. Shuttleworth, A. G. Todman, R. N. G. Naguib, B. M. Newman, and M. K. Bennett, "Colour texture analysis using co-occurrence matrices for classification of colon cancer images," in *Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering*, vol. 2, pp. 1134 – 1139, 2002.

[30] A. N. Esgiar, R. N. G. Naguib, B. S. Sharif, M. K. Bennett, and A. Murray, "Fractal analysis in the detection of colonic cancer images," *IEEE Transactions on Information Technology in Biomedicine*, vol. 6, no. 1, pp. 54 – 58, 2002.

[31] J. Filippas, H. Arochena, S. A. Amin, R. N. G. Naguib, and M. K. Bennett, "Comparison of two ai methods for colonic tissue image classification," in *Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2, pp. 1323 – 1326, 2003.

[32] A. G. Todman, R. N. G. Naguib, and M. K. Bennett, "Visual characterisation of colon images," *Proceedings of Medical Image Understanding and Analysis*, pp. 161 – 164, 2001.

[33] W. H. Wolberg, W. N. Street, D. M. Heisey, and O. L. Mangasarian, "Computer-derived nuclear features distinguish malignant from benign breast cytology," *Human pathology*, vol. 26, no. 7, pp. 792 – 796, 1995.

[34] J. P. Thiran and B. Macq, "Morphological feature extraction for the classification of digital images of cancerous tissues," *IEEE Transactions on Biomedical Engineering*, vol. 43, no. 10, pp. 1011 – 1020, 1996.

[35] P. W. Hamilton, P. H. Bartels, D. Thompson, N. H. Anderson, R. Montironi, and J. M. Sloan, "Automated location of dysplastic fields in colorectal histology using image texture analysis," *The Journal of Pathology*, vol. 182, no. 1, pp. 68 – 75, 1997.

[36] H. K. Choi, T. Jarkrans, E. Bengtsson, J. Vasko, K. Wester, P. U. Malmstrom, and C. Busch, "Image analysis based grading of bladder carcinoma. comparison of object, texture and graph based methods and their reproducibility," *Analytical Cellular Pathology*, vol. 15, no. 1, pp. 1 – 18, 1997.

[37] P. Spyridonos, D. Ravazoula, K. Cavouras, G. Berberidis, and P. Nikiforidis, "Computer-based grading of haematoxylin-eosin stained tissue sections of urinary bladder carcinomas," *Informatics for Health and Social Care*, vol. 26, no. 3, pp. 179 – 190, 2001.

[38] B. Nielsen, F. Albregtsen, and H. E. Danielsen, "The use of fractal features from the periphery of cell nuclei as a classification tool," *Analytical Cellular Pathology*, vol. 19, no. 1, pp. 21 – 37, 1999.

[39] S. J. Keenan, J. Diamond, W. Glenn McCluggage, H. Bharucha, D. Thompson, P. H. Bartels, and P. W. Hamilton, "An automated machine vision system for the histological grading of cervical intraepithelial neoplasia (cin)," *The Journal of Pathology*, vol. 192, no. 3, pp. 351 – 362, 2000.

[40] C. Gunduz-Demir, "Mathematical modeling of the malignancy of cancer using graph evolution," *Mathematical Biosciences*, vol. 209, no. 2, pp. 514 – 527, 2007.

[41] Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 800 – 810, 2001.

[42] H. D. Cheng and Y. Sun, "A hierarchical approach to color image segmentation using homogeneity," *IEEE Transactions on Image Processing*, vol. 9, no. 12, pp. 2071 – 2082, 2000.

[43] H. Mobahi, S. R. Rao, A. Y. Yang, S. S. Sastry, and Y. Ma, "Segmentation of natural images by texture and boundary compression," *International Journal of Computer Vision*, vol. 95, no. 1, pp. 86 – 98, 2011.

[44] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167 – 181, 2004.

[45] A. B. Tosun, M. Kandemir, C. Sokmensuer, and C. Gunduz-Demir, "Object-oriented texture analysis for the unsupervised segmentation of biopsy images for cancer detection," *Pattern Recognition*, vol. 42, no. 6, pp. 1104 – 1112, 2009.

[46] A. B. Tosun and C. Gunduz-Demir, "Graph run-length matrices for histopathological image segmentation," *IEEE Transactions on Medical Imaging*, vol. 30, no. 3, pp. 721 – 732, 2011.

[47] A. C. Simsek, A. B. Tosun, C. Aykanat, C. Sokmensuer, and C. Gunduz-Demir, "Multilevel segmentation of histopathological images using cooccurrence of tissue objects," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 6, pp. 1681 – 1690, 2012.

[48] A. B. Tosun, C. Sokmensuer, and C. Gunduz-Demir, "Unsupervised tissue image segmentation through object-oriented texture," in *Proceedings of the International Conference on Pattern Recognition*, pp. 2516 – 2519, 2010.

[49] M. M. Galloway, "Texture analysis using gray level run lengths," *Computer Graphics and Image Processing*, vol. 4, no. 2, pp. 172 – 179, 1975.

[50] World Health Organization, "Cancer," 2006.

[51] L. G. Shapiro and G. C. Stockman, *Computer Vision*. Prentice Hall, 2001.

[52] G. Mori, X. Ren, A. A. Efros, and J. Malik, "Recovering human body configurations: combining segmentation and recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 326 – 333, 2004.

[53] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1150 – 1157, 1999.

[54] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Adaptive perceptual color-texture image segmentation," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1524 – 1536, 2005.

[55] A. Ranganathan, "Semantic scene segmentation using random multinomial logit," in *Proceedings of British Machine Vision Conference*, 2009.

[56] L. Soh and C. Tsatsoulis, "Segmentation of satellite imagery of natural scenes using data mining," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 2, pp. 1086 – 1099, 1999.

[57] H. G. Akcay, S. Aksoy, and P. Soille, "Hierarchical segmentation of complex structures," in *Proceedings of the International Conference on Pattern Recognition*, pp. 1120 – 1123, 2010.

[58] P. Scheunders, "A genetic c-means clustering algorithm applied to color image quantization," *Pattern Recognition*, vol. 30, no. 6, pp. 859 – 866, 1997.

[59] T. Q. Chen and Y. Lu, "Color image segmentationan innovative approach," *Pattern Recognition*, vol. 35, no. 2, pp. 395 – 405, 2002.

[60] L. Shafarenko, H. Petrou, and J. Kittler, "Histogram-based segmentation in a perceptually uniform color space," *IEEE Transactions on Image Processing*, vol. 7, no. 9, pp. 1354 – 1358, 1998.

[61] E. Littmann and H. Ritter, "Adaptive color segmentation-a comparison of neural and statistical methods," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 175 – 185, 1997.

[62] H. D. Cheng, X. H. Jiang, and J. Wang, "Color image segmentation based on homogram thresholding and region merging," *Pattern Recognition*, vol. 35, pp. 373 – 393, 2002.

[63] Y. Xia, D. D. Feng, T. Wang, R. Zhao, and Y. Zhang, "Image segmentation by clustering of spatial patterns," *Pattern Recognition Letters*, vol. 28, no. 12, pp. 1548 – 1555, 2007.

[64] H. Wang and D. Suter, "Color image segmentation using global information and local homogeneity.," in *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications*, pp. 89 – 98, 2003.

[65] F. Kurugollu, B. Sankur, and A. E. Harmanci, "Color image segmentation using histogram multithresholding and fusion," *Image and Vision Computing*, vol. 19, no. 13, pp. 915 – 928, 2001.

[66] M. Mignotte, "Segmentation by fusion of histogram-based k-means clusters in different color spaces," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 780 – 787, 2008.

[67] M. Tuceryan and A. K. Jain, "Texture segmentation using voronoi polygons," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 211 – 216, 1989.

[68] W. Schwartz and H. Pedrini, "Texture classification based on spatial dependence features using co-occurrence matrices and markov random fields," in *Proceedings of the International Conference on Image Processing*, vol. 1, pp. 239 – 242, 2004.

[69] F. Y. Shih and S. Cheng, "Automatic seeded region growing for color image segmentation," *Image and Vision Computing*, vol. 23, no. 10, pp. 877 – 886, 2005.

[70] M. Krinidis and I. Pitas, "Color texture segmentation based on the modal energy of deformable surfaces," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1613 – 1622, 2009.

[71] D. K. Panjwani and G. Healey, "Markov random field models for unsupervised segmentation of textured color images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 939 – 954, 1995.

[72] B. Peng, L. Zhang, and D. Zhang, "Automatic image segmentation by dynamic region merging," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3592 – 3605, 2011.

[73] B. Russell, A. A. Efros, J. Sivic, B. Freeman, and A. Zisserman, "Segmenting scenes by matching image composites," in *Proceedings of the Advances in Neural Information Processing Systems*, December 2009.

[74] L. Shafarenko, M. Petrou, and J. Kittler, "Automatic watershed segmentation of randomly textured color images," *IEEE Transactions on Image Processing*, vol. 6, no. 11, pp. 1530 – 1544, 1997.

[75] A. Tremeau and N. Borel, "A region growing and merging algorithm to color segmentation," *Pattern Recognition*, vol. 30, no. 7, pp. 1191 – 1203, 1997.

[76] M. Celenk, "Hierarchical color clustering for segmentation of textured images," in *Proceedings of the Southeastern Symposium on System Theory*, pp. 483 – 487, 1997.

[77] S. Beucher and Centre De Morphologie Mathmatique, "The watershed transformation applied to image segmentation," in *Proceedings of the Scanning Microscopy International*, pp. 299 – 314, 1991.

[78] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888 – 905, 2000.

[79] J. S. Kim and K. S. Hong, "Color-texture segmentation using unsupervised graph cuts," *Pattern Recognition*, vol. 42, no. 5, pp. 735 – 750, 2009.

[80] Y.-W. Tai, J. Jia, and C.-K. Tang, "Soft color segmentation and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1520 – 1537, 2007.

[81] Z. Tu, K. Narr, P. Dollr, I. D. Dinov, P. M. Thompson, and A. W. Toga, "Brain anatomical structure segmentation by hybrid discriminative/generative models.," *IEEE Transactions on Medical Imaging*, vol. 27, no. 4, pp. 495 – 508, 2008.

[82] Z. Kato and T. C. Pong, "A markov random field image segmentation model for color textured images," *Image and Vision Computing*, vol. 24, no. 10, pp. 1103 – 1114, 2006.

[83] L. Zhang and Q. Ji, "Image segmentation with a unified graphical model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1406 – 1425, 2010.

[84] D. E. Ilea and P. F. Whelan, "Ctex - an adaptive unsupervised segmentation algorithm based on color-texture coherence," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1926 – 1939, 2008.

[85] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 2, pp. 142 – 159, 2000.

[86] R. Farjam, H. Soltanian-Zadeh, K. Jafari-Khouzani, and R. A. Zoroofi, "An image analysis approach for automatic malignancy determination of prostate pathological images," *Cytometry Part B: Clinical Cytometry*, vol. 72B, no. 4, pp. 227 – 240, 2007.

[87] C. Wittke, J. Mayer, and F. Schweiggert, "On the classification of prostate carcinoma with methods from spatial statistics," *IEEE Transactions on Information Technology in Biomedicine*, vol. 11, no. 4, pp. 406 – 414, 2007.

[88] S. Naik, S. Doyle, S. Agner, A. Madabhushi, M. D. Feldman, and J. Tomaszewski, "Automated gland and nuclei segmentation for grading of prostate and breast cancer histopathology," in *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 284 – 287, 2008.

[89] J. Kong, H. Shimada, K. Boyer, J. Saltz, and M. Gurcan, "Image analysis for automated assessment of grade of neuroblastic differentiation," in *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 61 – 64, 2007.

[90] K. Mosaliganti, F. Janoos, O. Irfanoglu, R. Ridgway, R. Machiraju, K. Huang, J. Saltz, G. Leone, and M. Ostrowski, "Tensor classification of n-point correlation function features for histology tissue segmentation," *Medical Image Analysis*, vol. 13, no. 1, pp. 156 – 166, 2009.

[91] N. Signolle, M. Revenu, B. Plancoulaine, and P. Herlin, "Wavelet-based multiscale texture segmentation: Application to stromal compartment characterization on virtual slides," *Signal Processing*, vol. 90, no. 8, pp. 2412 – 2422, 2010.

[92] C. Chen, J. A. Ozolek, W. Wang, and G. K. Rohde, "A general system for automatic biomedical image segmentation using intensity neighborhoods," *Journal of Biomedical Imaging*, pp. 1 – 12, 2011.

[93] A. Madabhushi, M. D. Feldman, D. N. Metaxas, J. E. Tomaszeweski, and D. Chute, "Automated detection of prostatic adenocarcinoma from high-resolution ex vivo mri," *IEEE Transactions on Medical Imaging*, vol. 24, no. 12, pp. 1611 – 1625, 2005.

[94] R. Bhagavatula, M. Fickus, W. Kelly, C. Guo, J. A. Ozolek, C. A. Castro, and J. Kovacandevic, "Automatic identification and delineation of germ layer components in he stained images of teratomas derived from human and nonhuman primate embryonic stem cells," in *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 1041 – 1044, 2010.

[95] E. Alpaydin, *Introduction to Machine Learning*. The MIT Press, 2004.

[96] A. Chu, C. M. Sehgal, and J. F. Greenleaf, "Use of gray value distribution of run lengths for texture analysis," *Pattern Recognition Letters*, vol. 11, no. 6, pp. 415 – 419, 1990.

[97] R. Haralick, "Statistical and structural approaches to texture," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 786 – 804, 1979.

[98] A. C. Simsek, "Multilevel cluster ensembling for histopathological image segmentation," Master's thesis, Computer Engineering Department, Bilkent University, Ankara, Turkey, 2011.

[99] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley India Pvt. Limited, 2001.